UNIVERSITAT DE LES ILLES BALEARS

Towards a gauge polyvalent numerical relativity code: numerical methods, boundary conditions and different formulations

per

Carles Bona-Casas

Tesi presentada per a l'obtenció
del títol de doctor

a la

Facultat de Ciències
Departament de Física

Dirigida per:
Prof. Carles Bona Garcia i Dr. Joan Massó Bennàssar

"The fact that we live at the bottom of a deep gravity well, on the surface of a gas covered planet going around a nuclear fireball 90 million miles away and think this to be normal is obviously some indication of how skewed our perspective tends to be."

Douglas Adams.

Acknowledgements

I would like to acknowledge everyone who ever taught me something. Specially my supervisors who, at least in the beginning, had to suffer my eyes, puzzling at their faces, saying that I had the impression that what they were explaining to me was some unreachable knowledge. It turns out that there is not such a thing and they have finally managed to make me gain some insight in this world of relativity and computing. They have also infected me the disease of worrying about calculations and stuff that most people wouldn't care about and my hair doesn't seem very happy with it, but I still love them for that.

Many thanks to everyone in the UIB relativity group and to all the PhD. students who have shared some time with me. Work is less hard in their company. Special thanks to Carlos, Denis, Sasha and Dana for all their very useful comments and for letting me play with their work and codes. I would also like to thank everyone at the AEI for their hospitality during my stay. Specially Barry Wardell, who was a great flatmate and is a better person. I wouldn't like to forget about the organisers and assistants of the Spanish Relativity Meetings, many of them having become friends (Lode, Jen, Isabel, Jose...).

And, to conclude, to my family for their support and comprehension, to my friends for helping me avoid madness, to Lorena, the best of the office mates ever, to the Spanish Government for funding and above all to my girlfriend, Marta, who is always there no matter what.

Contents

Als meus pares i a na Marta

Preface

During the past years there have been intense research efforts on black holes and their effect on the astrophysical environment, and specially for the last six years, one of the systems which has drawn most attention is a binary system formed of two black holes. Now we know that there are three phases in the coalescence of two black holes: the inspiral phase, when the black holes are far from each other; the merger phase, when they are significantly closer and the system becomes highly non-linear; and the ringdown phase, after the two holes have merged leaving a single black hole in an excited state emitting radiation.

Coalescences of two black holes are astrophysical events that release great amounts of energy in the form of gravitational radiation and, given the case of supermassive black holes, in the form of dual jets too [1]. In fact, the final merger of two black holes in a binary system releases more power (in gravitational waves) than the combined light from all the stars in the visible Universe (in photons) [2]. This energy that comes in the form of gravitational waves travels across the Universe at the speed of light and carries the waveform signature of the merger.

Events that release such an outstanding amount of energy are key sources for gravitational-wave detectors. In fact, they are one of the most likely sources for the first detection. But despite the energy released, as gravity is the weakest of the fundamental forces, the output of ground-based detectors is dominated by different kind of noise sources: thermal noise (heating of the antennae instruments), seismic noise (even though for example mirrors are suspended in vacuum chambers) and shot noise (the statistical error that comes from taking averages over a number of photons received at the photodetector). As a consequence, sophisticated statistical algorithms must be used in order to extract physical signals corresponding to the detection of gravitational waves from binary black hole systems. These algorithms require accurate waveform templates that correspond to the sources that are to be detected.

Calculating these waveforms requires solving the full Einstein equations of general relativity on a computer in three spatial dimensions plus time. Numerical relativists have attempted to solve this problem for many years, but they were faced with a number of instabilities that made their numerical codes crash before they could compute any sizable portion of a binary orbit. Remarkably, in the past few years a series of dramatic breakthroughs has occurred in numerical relativity (NR), yielding robust and accurate simulations of black-hole mergers for the first time. Numerical solutions of Einstein's equations for the last orbits and merger of a black-hole binary, the ringdown of the single black hole that remains, and the GWs emitted in the process, became possible in 2005 [3–6]. Since that time many simulations have been performed, but they all share some common grounds and techniques.

Astrophysical black holes ultimately form through gravitational collapse of matter, but in a black-hole simulation one does not need describe this process at all. The black hole can instead be represented purely through its effect on the spacetime geometry. The spacetime singularity at the center of a black hole is difficult to describe numerically, and there are different approaches to this problem. In the excision technique, which was first proposed in the late 1980s [7], a portion of a spacetime inside of the event horizon surrounding the singularity of a black hole is simply not evolved. In theory this should not affect the solution to the equations outside of the event horizon because of the principle of causality and properties of the horizon (i.e. nothing physical inside the black hole can influence any of the physics outside the horizon). This is, of course, if we don't take into account quantum tunneling, which is at the origin of Hawking's radiation. Thus if one simply does not solve the equations inside the horizon one should still be able to obtain valid solutions outside. One "excises" the interior by imposing ingoing boundary conditions on a boundary surrounding the singularity but inside the horizon. While the implementation of excision has been very successful, the technique has two problems. The first is that one has to be careful about the coordinate conditions. Although physical information cannot escape the black-hole, non-physical numerical or gauge information can in principle escape, and may lead to numerical instabilities. The second problem is that as the black holes move, one must continually adjust the location of the excision region to move with the black hole. Excision is used in the pioneering Pretorius code [3, 8, 9], and in the `SpEC` code [10]. Pretorius's original simulations began with scalar-field initial data, chosen such that it would quickly collapse to form a black hole. Once the black hole had formed, the interior (and the remaining scalar field) were excised.

Another method of avoiding singularities is to choose coordinates that bypass them: the black holes are initially described with topological wormholes, such that as the numerical coordinates approach one of the black holes, they pass through a wormhole and instead

of getting closer to the singularity end up further away, in a new asymptotically flat region. A coordinate transformation is performed to compactify these wormholes, and the extra asymptotically flat regions are reduced to single points, called punctures [11–14]. Until 2005, all published usage of the puncture method required that the coordinate position of all punctures remain fixed during the course of the simulation. Of course black holes in proximity to each other will tend to move under the force of gravity, so the fact that the coordinate position of the puncture remained fixed meant that the coordinate systems themselves became "stretched" or "twisted," and this typically lead to numerical instabilities at some stage of the simulation. In 2005 some research groups demonstrated for the first time the ability to allow punctures to move through the coordinate system, thus eliminating some of the earlier problems with the method. This 'moving puncture' approach represented also a breakthrough that allowed accurate long-term evolutions of black holes in the puncture approach [4–6].

A third option could be a scalar field stuffing, which for some reason is not yet used in binaries. We have mentioned that Pretorius was using it in his original simulations but then the interior was excised after collapse. Here we refer to the possibility of evolving a binary black hole without either puncture-like initial data or excision at all, but apparently there is a tight bond between the type of initial data and the formalism used. Punctures are unavoidably associated with the BSSN system whereas excision is used solely in harmonic formulations.

We have just mentioned two different formulations of the Einstein equations. Given black-hole-binary initial data, a stable evolution requires a numerically well-posed and stable formulation of Einstein's equations, as well as a specific choice of gauge conditions. Finding a suitable set of evolution equations and gauge conditions was one of the major problems in the field during the decade preceding the 2005 breakthroughs. Although not all mathematical and numerical questions have been resolved, long-term stable simulations can now be performed with either a variant of the generalized harmonic formulation [8, 15–17] or the moving-puncture treatment [4–6] of the Baumgarte-Shapiro-Shibata-Nakamura (BSSN) [18, 19] formulation.

Harmonic formalisms originated with consideration of "harmonic coordinates", so called because the coordinates satisfy the wave equation $\Box x^\mu = 0$, where the box stands for the general-covariant wave operator acting on functions. In these coordinates, Einstein's equations can be written such that the principal part resembles a wave equation in terms of the metric:

$$\Box\, g_{ab} = \;\cdots\; - 16\,\pi\,(T_{ab} - \frac{T}{2}\,g_{ab})\,, \tag{1}$$

Where the dots stand for terms quadratic in the metric first derivatives. In this form, Einstein's equations are manifestly hyperbolic [20]. However, the harmonic coordinate condition is too restrictive for numerical purposes, so generalized harmonic coordinates were eventually developed by introducing a source term into the coordinate condition, i.e. $\Box x^\mu = H^\mu$ [15, 21], a suitable choice for which preserves strong hyperbolicity. The subsequent introduction of *constraint-damping* terms, which tend to drive the constraints towards zero, further ensured stability [22]. This formulation is manifestly second-order in both time and space, and has been implemented numerically as such [9], but for more efficient numerical integration a first-order-in-time formulation was also developed [17], and is currently being used by some groups..

The BSSN decomposition starts instead with the (numerically ill-posed) ADM-York equations for the spatial quantities (γ_{ij}, K_{ij}) [23, 24]. The BSSN reformulation provides evolution equations for conformally rescaled quantities, $\{\psi, K, \tilde{\gamma}_{ij}, \tilde{A}_{ij}, \tilde{\Gamma}^i\}$, where $\gamma_{ij} = \psi^4\tilde{\gamma}_{ij}$ and $K_{ij} = \psi^4(\tilde{A}_{ij} + \tilde{\gamma}_{ij}K)$, and the extra variable, $\tilde{\Gamma}^i = \partial_j\tilde{\gamma}^{ij}$ is introduced. The moving-puncture extension of the BSSN system deals with puncture data, and involves introducing either $\phi = \ln\psi$ [5], $\chi = \psi^{-4}$ [4] or $W = \psi^{-2}$ [25], and evolving that quantity instead of the conformal factor ψ, and specifying gauge conditions that allow the punctures to move across the numerical grid.

The choice of gauge or coordinate conditions, like the choice of formulation, has important consequences on the numerics, especially the stability of the simulation. Important considerations include how to deal with the extreme conditions of black holes such as the physical singularities, the possible coordinate singularities, the strong-field gradients, and the dynamical, surrounding spacetime. The coordinates must accommodate these features in a way that is numerically tractable.

In particular, BSSN deals with the 1+log slicing of the Bona-Massó family [26] together with

$$\partial_t\beta^i = \frac{3}{4}\tilde{\Gamma}^i + \beta^j\partial_j\beta^i - \eta_\beta\beta^i. \tag{2}$$

Where $\tilde{\Gamma}^i = -\partial_j\tilde{\gamma}^{ij}$ depends on a conformal three-metric $\tilde{\gamma}_{ij}$ of the evolving spatial slice and β^i is the shift. η_β is a damping parameter that fine-tunes the growth of the shift, which affects the coordinate size of the black-hole horizons, which in turn has bearing on the required numerical resolution [27, 28]. Use of this or similar gauge conditions has become known as the "moving puncture" method, and proved to be very successful as it has become increasingly widespread among the numerical-relativity community.

Development of generalized harmonic coordinates initially proceeded independently of the above 3+1-formulated conditions. As mentioned, in harmonic coordinates the D'Alembertian of each coordinate vanishes. In generalized harmonic coordinates, the wave equation for each coordinate is allowed a source term, i.e.

$$\Box x^\mu = H^\mu. \tag{3}$$

These "gauge driving" source terms H^μ can be either algebraically specified or evolved such that hyperbolicity is preserved [9, 15, 17, 21].

The first successful numerical orbit of black holes involved a source term for the time coordinate that effectively kept the lapse close to its Minkowski value of unity, while the spatial coordinates remained harmonic [9]. This was accomplished by evolving the source term itself, according to

$$\Box H_0 = \left[-\xi_1(\alpha - 1) + \xi_2(\partial_t - \beta^i\partial_i)H_0\right]\alpha^{-1} \tag{4}$$

where ξ_1 and ξ_2 are constants. More recently, to dampen extraneous gauge dynamics during the inspiral and merger of generic binaries, [29] found the following gauge driver to be successful:

$$H_0 = \mu_0\left[\log\left(\frac{\sqrt{g}}{\alpha}\right)\right]^3 \tag{5}$$

$$H_i = -\mu_0\left[\log\left(\frac{\sqrt{g}}{\alpha}\right)\right]^2\frac{\beta_i}{\alpha} \tag{6}$$

where μ_0 is a specified function of time that starts at zero and eventually increases monotonically to unity.

The generalized-harmonic and moving-puncture methods have been found to work for simulations of up to 15 orbits, for binaries with significant eccentricity, with mass ratios up to 1:10, and spins up to the conformal-flatness limit of $a/m \sim 0.93$. Despite this wealth of evidence that these methods work, surprisingly little has been done to explain why. The properties that are known to be necessary for a stable simulation (in particular, a strongly hyperbolic evolution system), are also known to not be sufficient. What distinguishes these methods from others? Could it be that most other (well-posed) systems of equations can be stably evolved with appropriate gauge conditions and methods to move the black holes through the grid? Why BSSN is so successful at simulating black holes but fails tests such as the gauge waves test? These questions have been largely neglected, and deserve more attention.

To accurately simulate a binary black hole spacetime, a computer code must adequately resolve both the region near the black hole, and the spacetime far away, where gravitational waves are extracted. However, if one had to resolve the whole domain with very high resolution with no exception, it would be faced with a lack of computational memory to store all the information. Luckily enough, the resolution needed in the gravitational wave extraction area is fairly below the one needed near the black holes. To deal with such large differences in resolution requirements on different parts of the computational domain many codes use mesh refinement methods [30]. Another technique is to use a coordinate transformation that changes the effective resolution in different regions; such a "fisheye" transformation was used in early results from the `LazEv` code [4, 31–34], and was also used in more recent simulations by the UIUC group, for example [35]. A third option is to divide the computational domain into a number of different domains or patches, and use a different numerical resolution and even different coordinate systems in each domain; a multi-domain method is used in the `SpEC` code [10] and in the `Llama` code [36, 37]

Both the numerical and physical accuracy of numerical waveforms has improved steadily since 2005. The first simulations were performed with a code that resolved each time slice with second-order-accurate finite differences [3]. The moving-puncture results that followed six months later [4, 5] used second- and fourth-order-accurate finite differences. An accurate comparison of numerical and post-Newtonian waveforms was performed in 2007 using sixth-order finite-differencing , and the `LazEv` code now routinely uses eighth-order methods . The `SpEC` code, which has produced the most accurate equal-mass nonspinning binary waveform to date, uses pseudospectral methods to describe the spatial slice.

Ideally the outer boundary of the computational domain is located at spatial or null infinity. The only long-term binary evolution code where one of these techniques is employed is that of Pretorius, where spatially compactified coordinates are used [3, 8]. The region near the outer boundary is by definition poorly resolved, but a filtered buffer zone between the well- and poorly-resolved regions is used to reduce the build-up and propagation of any resulting errors. In all other codes the outer boundary of the computational domain is not at spatial infinity, and boundary conditions must be imposed. The physically correct outer boundary conditions are not known for a black-hole-binary spacetime, so one has to provide some alternative. Ideally, boundary conditions should result in a solution which is indistinguishable from an evolution with an infinite spatial domain. This can only be achieved approximately, but still the boundary conditions should have certain properties in order to give a useful solution. Firstly, they should not contaminate the solution with unphysical gravitational radiation, either due to reflections of the waves generated by the simulated system, or due to radiation generated by

the boundary condition itself. Secondly, they should be constraint preserving to yield a result which is a solution to the Einstein equations. Thirdly, they should result in a well-posed initial boundary value problem. This is a mathematical property which is a necessary condition for the numerical schemes used to be formally stable. These three properties are often only approximately satisfied, as for example the BSSN codes generally use Sommerfeld-like outer boundary conditions (which are physically correct only for a spherically symmetric wave pulse on a flat background), and the outer boundary is placed as far from the binary system as computational resources allow. The Caltech-Cornell `SpEC` code uses a set of constraint-preserving boundary conditions that provide a far better approximation to the correct physics of outgoing waves on a dynamical spacetime than Sommerfeld conditions, and make it possible to place the outer boundary closer and still achieve accurate results .

These simulations require large computational resources. Long black-hole-binary simulations are typically run on multiple processors of a supercomputer, and we can get an impression of the "size" of a simulation from the amount of memory it requires, and the number of CPU hours it takes to run. As an example, a high-accuracy equal-mass nonspinning waveform can take roughly 18 days running on 24 processors, for a total of about 10,000 CPU hours.

With all these elements on the table, the simulation of binary black holes has been possible and very fruitful. Even though, after the initial gold rush with research groups competing for more orbits, higher mass ratios and spin of the black holes, the exploration into new regions of parameter space has now slowed significantly and there are some important points at the fundamental level that have been left behind. In this thesis, we would like to present some works we have carried out in this direction, trying to answer some questions or improving some aspects that at some point were neglected for the sake of obtaining gravitational wave patterns at any cost.

As we have said before, many research groups use finite differencing with some sort of artificial viscosity to overcome some junk radiation present in the initial data and also created by numerical effects of steep slopes and mesh interpolation. On top of that mesh refinement is used, and very specific gauge choices that freeze the growth of the black hole horizon are needed so that the code does not have to deal with very strong field gradients and complex dynamics. So, to start, we will present a new finite volume method in the context of numerical relativity. Finite volume methods were developed by the fluid dynamics community and they have been widely tested and have developed a well-deserved reputation of robustness. A reputation that finite difference methods certainly lack. On the contrary, finite volume methods are sometimes regarded as inefficient because they need the full characteristic decomposition of the system. This

is not true anymore, there are finite volume methods that only need the eigenvalues of the system and they use a flux formula. This is why we will present a method that does not need the full characteristic decomposition of the equations and we will use it to successfully perform some numerical relativity simulations with the Z3 formalism, developed at UIB, in Chapter 1.

Even though, one might still wonder why these finite difference methods are so successful in numerical relativity. If in the first chapter we used the fluid dynamics language, in the second chapter we will try to travel back to the finite difference context: we will try to compare the obtained method with the ones used in numerical relativity. By doing so we will find many similarities and incidentally a very efficient implementation of the method presented in the first chapter. We will also find that, with a very small modification, we can generalise our method to a whole family of methods and find some experimental proof of their robustness by performing some fluid physics tests with results being published in Journal of Computational Physics. We must say it is remarkable that these methods have been developed in the numerical relativity context and are being used in hydrodynamics calculations [38, 39] and not the other way round.

In Chapter 3 we will break an existing contradiction in the numerical relativity community. If general relativity possesses freedom of choice regarding gauge conditions as an important feature of the theory, numerical relativity does not. Usually the argument is used in reverse to defend the results: if we can run our simulations with a single gauge condition, then the theory ensures we will not find anything new by changing it. But still it is puzzling that the existent numerical relativity codes rely on very specific gauge choices as we have mentioned earlier. With the Z4 formalism, developed at UIB, we obtain some unprecedented flexibility in this regard: with the help of the numerical methods presented in Chapter 2, we are able to evolve a 3-dimensional black hole in normal coordinates (something which none of the preexistent formalisms could do) with a cartesian grid, with regular initial data (scalar field stuffing), without mesh refinement and, more importantly, without the gauge choice being a specific requisite as we are able to perform simulations with different shift conditions and different slicing conditions too.

Both BSSN and the Generalised Harmonic formalisms are free evolution formalisms. This means that both energy and momentum constraints are ensured with some compatible initial data. But this is only at the continuum level. In numerical simulations one does need boundary conditions, and if they don't preserve the constraints our solutions can go to a solution space that includes Einstein but might not be Einstein. The standard practice is to place the boundaries very far away, oping that this will not affect our domain of interest. We develop instead in Chapter 4 a set of constraint preserving boundary conditions and we show their effectiveness by implementing them even in the

strong field regime (with unprecedented results) and in 3 dimensions in cartesian-like grids including corners.

In Chapter 3 we perform a numerical test that shows that Z4 can be much more accurate than BSSN. More recently this has been confirmed by the work of others [40]. Therefore, by the end of the thesis we have made an effort to use a second order system, puncture initial data and mesh refinement (same as BSSN) and we show in Chapter 5 some preliminary results were it seems like it is plausible that Z4 can work under these conditions.

And, finally, as a thesis is a long term project, one does find unexpected things in the way. None of the nowadays used Einstein generalisations in numerical relativity had ever been derived from an action principle. We see how this can be accomplished with the Palatini approach in Chapter 5. This opens many ways both at the theoretical level (with interest in quantum gravity theories) and at the numerical level with the use of numerical (symplectic) methods that exactly preserve the constraints during the evolution. This finding can be regarded as an unexpected theoretical landmark.

References

[1] Carlos Palenzuela, Luis Lehner, and Steven L. Liebling Science 20 August 2010: 329 (5994), 927-930.

[2] Ralph A.M.J. Wijers Mon. Not. R. Astron. Soc. 000, 12 (2005)

[3] Pretorius F 2005 *Phys. Rev. Lett.* **95** 121101

[4] Campanelli M, Lousto C O, Marronetti P and Zlochower Y 2006 *Phys. Rev. Lett.* **96** 111101

[5] Baker J G, Centrella J, Choi D I, Koppitz M and van Meter J 2006 *Phys. Rev. Lett.* **96** 111102

[6] Diener P, Herrmann F, Pollney D, Schnetter E, Seidel E, Takahashi R, Thornburg J, Ventrella J. 2006 *Phys. Rev. Lett.* **96** 121101

[7] Thornburg J 1987 *Class. Quantum Grav.* **4**(5) 1119–1131

[8] Pretorius F 2005 *Class. Quantum Grav.* **22** 425–452

[9] Pretorius F 2006 *Class. Quantum Grav.* **23** S529–S552

[10] Scheel M A *et al.* 2006 *Phys. Rev.* **D74** 104006

[11] Beig R and O'Murchadha N 1994 *Class. Quantum Grav.* **11** 419

[12] Beig R and Husa S 1994 *Phys. Rev. D* **50** R7116–7118

[13] Dain S and Friedrich H 2001 *Comm. Math. Phys.* **222** 569

[14] Brandt S and Brügmann B 1997 *Phys. Rev. Lett.* **78**(19) 3606–3609

[15] Friedrich H 1985 *Comm. Math. Phys.* **100** 525–543

[16] Friedrich H and Rendall A D 2000 *Lect. Notes Phys.* **540** 127–224

[17] Lindblom L, Scheel M A, Kidder L E, Owen R and Rinne O 2006 *Class. Quantum Grav.* **23** S447–S462

[18] Shibata M and Nakamura T 1995 *Phys. Rev. D* **52** 5428

[19] Baumgarte T W and Shapiro S L 1998 *Phys. Rev. D* **59** 024007

[20] Choquet-Bruhat, Y. The cauchy problem. In, Gravitation: an Introduction to Current Research, 130168. Wiley, New York.

[21] Garfinkle, D. *Phys. Rev. D* **65** 044029

[22] Gundlach C., Calabrese G., Hinder I. and Martin-Garcia J M *Class. Quantum Grav.* **22** 3767–3774

[23] Arnowitt R, Deser S and Misner C W 1962 in L Witten, ed, Gravitation an introduction to current research (John Wiley, New York) pp 227265

[24] York J W 1979 in L L Smarr, ed, Sources of gravitational radiation (Cambridge, UK: Cambridge University Press) pp 83126 ISBN 0-521-22778-X

[25] Marronetti P, Tichy W, Brügmann B, González J and Sperhake U 2008 Phys. Rev. D **77** 064010

[26] Bona C, Massó J, Seidel E, Stela J. Phys. Rev. Lett. 75 (1995), 600

[27] González, José A. and Sperhake, Ulrich and Brügmann, Bernd Phys.Rev.D 79 124006 (2009)

[28] Brügmann, Bernd; González, José A.; Hannam, Mark; Husa, Sascha; Sperhake, Ulrich; Tichy, Wolfgang Phys. Rev. D 77, 024027 (2008)

[29] Béla Szilágyi, Lee Lindblom, and Mark A. Scheel. Phys. Rev. D 80, 124010 (2009)

[30] Berger M J and Oliger J 1984 *J. Comput. Phys.* **53** 484–512

[31] Campanelli M, Lousto C O and Zlochower Y 2006 *Phys. Rev. D* **73** 061501

[32] Campanelli M, Lousto C O and Zlochower Y 2006 *Phys. Rev. D* **74** 041501

[33] Campanelli M, Lousto C O and Zlochower Y 2006 *Phys. Rev. D* **74** 084023

[34] Campanelli M, Lousto C O, Zlochower Y and Merritt D 2007 *Astrophys. J.* **659** L5–L8

[35] Etienne Z B, Faber J A, Liu Y T, Shapiro S L and Baumgarte T W 2007 *Phys. Rev.* **D76** 101503

[36] Denis Pollney, Christian Reisswig, Nils Dorband, Erik Schnetter, Peter Diener Phys. Rev. **D80**, 121502 (2009)

[37] Denis Pollney, Christian Reisswig, Erik Schnetter, Nils Dorband, and Peter Diener Phys. Rev. **D83**, 044045 (2011)

[38] S. Rial, I. Arregui, J. Terradas, R. Oliver and J. L. Ballester 2010 ApJ **713** 651

[39] M. Luna, J. Terradas, R. Oliver and J. L. Ballester 2010 ApJ **716** 1371-1380

[40] M. Ruiz, D. Hilditch and S. Bernuzzi, Phys. Rev. D**83** 024025 (2011)

Chapter 1

Centered FV Methods

1.1 Introduction

Let us consider now the well known 3+1 decomposition of Einstein's field equations.

$$
\begin{aligned}
ds^2 &= -(\alpha^2 - \beta^i\beta_i)dt^2 + 2\beta_i dx^i dt + \gamma_{ij}dx^i dx^j && (1.1)\\
(\partial_t - \mathcal{L}_\beta)\ \gamma_{ij} &= -2\alpha K_{ij} && (1.2)\\
(\partial_t - \mathcal{L}_\beta)\ K_{ij} &= -\nabla_i\alpha_j + \alpha \Big[\ R_{ij} - 2K_{ij}^2 + trK\ K_{ij} && (1.3)\\
&\quad -S_{ij} + \frac{1}{2}(trS - \tau)\gamma_{ij}\ \Big] && (1.4)
\end{aligned}
$$

Where we have only written the line element and the evolution equations, omitting the energy-momentum constraints. R_{ij} are the components of the Ricci tensor, S_{ij} are the space components of the stress-energy tensor and τ is the energy density. The extrinsic curvature K_{ij} is considered as an independent dynamical field, so that the evolution system is of first order in time but second order in space. Let us transform it into a fully first order system by considering also the first space derivatives of the metric as independent quantities. This requires additional evolution equations for these space derivatives, that can be obtained in the standard way by permuting space and time derivatives of the metric, that is

$$\partial_t\ (\partial_k\ g_{ab}) = \partial_k\ (\partial_t\ g_{ab})\ , \tag{1.5}$$

so that the resulting first order system will describe the same dynamics than the original second order one.

In this first order form, Einstein's field equations can always be expressed as a system of balance laws [1]. The evolution system can be written in the form

$$\partial_t \ \mathbf{u} + \partial_k \ \mathbf{F}^k(\mathbf{u}) = \mathbf{S}(\mathbf{u}) \ , \tag{1.6}$$

where both the Flux terms **F** and the Source terms **S** depend algebraically on the array of dynamical fields **u**, which contains the metric and all its first derivatives. The terms 'Fluxes' and 'Sources' come from the hydrodynamical analogous of the system (1.6).

The balance law form is specially suited for the Method of Lines (MoL) discretization. Many current BH simulations are performed with the MoL technique. The MoL is the generic name of a family of discretzation methods in which time and space variables are dealt with separately. This is in keeping with the 3+1 framework, where the natural way of time discretization is by finite differences (FD) whereas one would like to keep all the options open for space discretization: finite differences, finite volume or even spectral methods.

To illustrate the idea, let us consider a 'semi-discrete' system in which only the time coordinate is discretized, whereas space derivatives are kept at the continuum level. The evolution of the array **u** of dynamical fields is written as

$$\partial_t \ \mathbf{u} = \mathbf{RHS} \ , \tag{1.7}$$

where the right-hand-side array **RHS** contains the remaining terms in the evolution equations, including the space derivative ones. In this way, we are disguising in (1.7) the original system of partial differential equations (PDE) as a system of ordinary differential equations (ODE), assuming that we will manage to compute the right-hand-side term **RHS** at every level, but ignoring for the moment the details.

This 'black box' approach allows us to apply the well-known ODE discretization techniques to get the required time resolution, using the Euler step (forward time difference)

$$\mathbf{u}^{(n+1)} = \mathbf{u}^{(n)} + \Delta t \ \mathbf{RHS}(t_n, \mathbf{u}^{(n)}) \ , \tag{1.8}$$

as the basic building block for advanced multi-step methods, like the modified-midpoint or Runge-Kutta algorithms [2, 3]. For more details on the time discretization used, please see Appendix B.

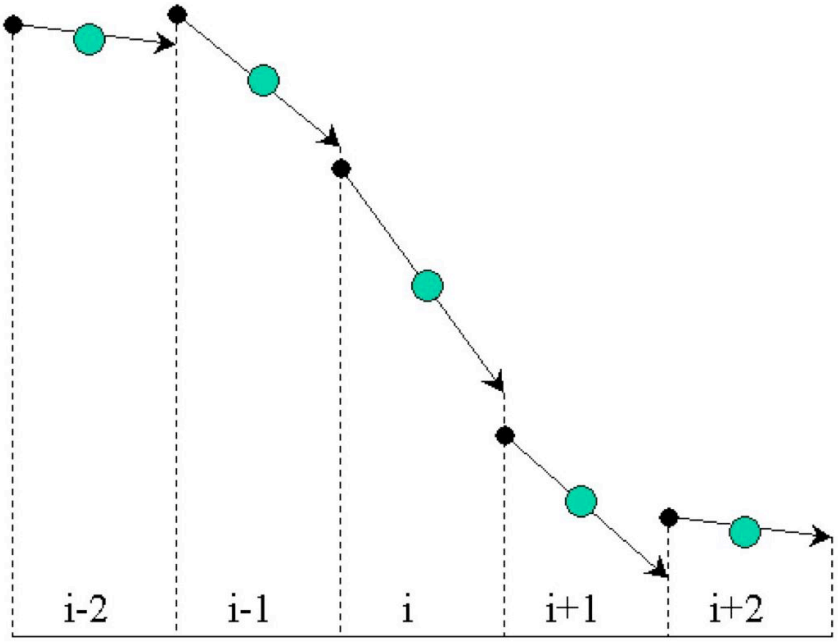


FIGURE 1.1: Piecewise linear reconstruction of a given function. Numerical discontinuities appear at every cell interface (dotted lines) between the left and right values (arrows and dots, respectively). Note that the original function was monotonically decreasing: all the slopes are negative. However, both the left interface values (at $i+3/2$) and the right interface ones (at $i-3/2$) show local extremes that break the monotonicity of the original function.

As in MoL there is a clear-cut separation between space and time discretization, the source terms contribute in a trivial way to the space discretization. The non-trivial contribution comes just from the flux-conservative part.

The balance law form is also well suited for FV discretization methods. The idea is to evolve the average of the dynamical fields $\mathbf{u}$ on some elementary cells, instead of evolving just point values like in the FD approach. The space discretization can be obtained by averaging (1.6) over an elementary cell and applying the divergence theorem to get:

$$\partial_t \, \bar{\mathbf{u}} + \oint \mathbf{F}^k \, dS_k = \bar{\mathbf{S}} \,, \tag{1.9}$$

where the overlines stand for space averages. The evaluation of partial space derivatives has been replaced in this way by that of surface integrals of the flux terms.

The idea behind Finite Volume (FV) methods, as we have seen, is to evolve the average of the dynamical fields over elementary cells instead of evolving only values at a single point, as in the Finite Difference (FD) methods. These values are modified at each timestep using the flux that goes through the interfaces of the cells of the grid; and so finding suitable functions for numerical fluxes to approximate these fluxes correctly becomes the primary problem. These numerical fluxes are, in general, functions of the

state of the system at each side of the interface. And to find the state at each side of the interface one usually needs to reconstruct the original function departing from the only available data: the average of the field at the cell.

Let us consider for simplicity the one-dimensional case. We can start from a regular finite difference grid. The elementary cell can then be chosen as the interval $(x_{i-1/2}\,,\ x_{i+1/2})$, centered on the generic grid point x_i. The dynamical fields $\mathbf{u}$ can be modelled as piecewise linear functions in every cell (linear reconstruction, see Fig. 1.1), so that the average values $\bar{\mathbf{u}}_i$ coincide with the point values $\mathbf{u}_i$. The corresponding FV discretization of (1.9) is then given by

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n \quad - \quad \frac{\Delta t}{\Delta x}\,[\ \mathbf{F}^x_{i+1/2} - \mathbf{F}^x_{i-1/2}\ ] + \Delta t\ \mathbf{S}_i\,. \tag{1.10}$$

We will restrict ourselves to these linear reconstruction methods in the following sections, but a more basic solution would be to use only these averages for the reconstruction. That is, the reconstruction process would be the simplest one possible: we approximate our original function by a piecewise constant function. Then we try to find information about the fluxes solving the Riemann problem. The Riemann problem consists in nothing else but solving the hyperbolic PDE with some special initial data. The initial data, given by the chosen reconstruction, are, in this case, piecewise constant with a step discontinuity at some point, for example x=0,

$$q(x,0) = \begin{cases} q_l & \text{if } x < 0 \\ q_r & \text{if } x > 0 \end{cases} \tag{1.11}$$

where q_l and q_r are the values from the left and from the right respectively. If we have the averages of two neighbouring cells on a finite volume grid, we can interpret the numerical discontinuity that they form as a physical Riemann problem that can be solved to obtain information that allows us to calculate a numerical flux and therefore updating the averages of the cells after a timestep.

This basic approach, called the Godunov [4] approach, gives us a first order approximation only. There have been many modifications to this approach with the goal of obtaining a higher precision, for example using a linear or parabollic reconstruction instead of a constant one. But the vast majority keep solving the Riemann problem at every interface of every cell at each timestep, mainly because these methods are thought to perform simulations that do have step discontinuities. This has given FV methods a reputation of being computationally expensive, a price that is not worth to pay for spacetime simulations, where the dynamical fields usually have smooth profiles. This reputation comes from the fact that, in order to solve the Riemann problem, one needs a

spectral decomposition of the Jacobian matrix of the system at each interface of each cell at each timestep, with the important computational cost implied by these calculations. In multidimensional simulations it is common to use a technique called dimensional separation, which requieres knowing eigenvalues and eigenvectors at each interface for each of the dimensions. Computational cost skyrockets: with a modest bidimensional grid of 100x100 cells, for example, one has to solve a minimum of 20000 Riemann problems at each timestep to implement the most simple generalization of the Godunov method in 2 dimensions.

From this point of view, centered FV methods can provide some improvement, because they do not require the full characteristic decomposition of the set of dynamical fields: only the values of the propagation speeds are needed [4].

This point can be illustrated by comparing the classical FV techniques implemented in a previous work at the UIB [5] with the new FV methods presented in this chapter. In [5], the general relativistic analogous of the Riemann problem must be solved at every single interface. This implies transforming back and forth between the primitive variables (the ones in which the equations are expressed) and the characteristic ones (the eigenvectors of the characteristic matrix along the given axis). In the present chapter, as we will see in next section, a simple flux formula is applied directly on the primitive variables, so that switching to the characteristic ones is no longer required. The flux formula requires just the knowledge of the characteristic speeds, not the full decomposition.

Another important difference is that in [5], the primitive quantities where reconstructed from their average values in a piecewise linear way, using a unique slope at every computational cell. Only (piecewise) second order accuracy can be achieved in this way, so that going to (piecewise) third order would require the use of 'piecewise parabolic methods' (PPM), with the corresponding computational overload. Here instead we will split every flux into two components before the piecewise-linear reconstruction (flux-splitting approach [4]). This will allow using a different slope for every flux component: this extra degree of freedom will allow us to get (piecewise) third order accuracy for a specific choice of slopes, without using PPM.

It is true that third-order convergence is rarely seen in practice. In the context of Computational Fluid Dynamics (CFD), this is due to the arising of physical solutions (containing shocks or other discontinuities) which are just piecewise smooth. These discontinuities can propagate across the computational domain and the convergence rate is downgraded as a result in the regions swept away by the discontinuity front. A similar situation is encountered in black hole evolutions. The use of singularity avoidant slicing conditions produces a collapse in the lapse function. As it can be seen in Fig. 1.2, a steep gradient surface is formed (the collapse front) that propagates out as the grid

points keep falling into the black hole. We will see that this results into a downgrade of accuracy in the regions close to the collapse front.

Stability problems can also arise from the lack of resolution of the collapse front, which is typically located around the apparent horizon. The reconstruction procedure can lead there to spurious oscillations, which introduce high-frequency noise in the simulation. In [5], this problem was dealt with the use of standard slope limiters, which were crucial for the algorithm stability. In the present chapter, although slope limiters are also discussed for completeness, their use is not even required in any of the presented simulations. The new algorithm gets rid by itself of the high-frequency noise, even for the steep (but smooth) profiles appearing around the black-hole horizon.

1.2 Flux formulae

The generic algorithm (1.10) requires some prescription for the interface fluxes $\mathbf{F}^x_{i\pm1/2}$. A straightforward calculation shows that the simple average

$$F_{i+1/2} \;=\; \frac{1}{2}\,(F_i + F_{i+1}) \tag{1.12}$$

And therefore

$$F_{i-1/2} \;=\; F_{i-1+1/2} \;=\; \frac{1}{2}\,(F_{i-1} + F_i) \tag{1.13}$$

In combination with (1.10) gives

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n \;\; - \;\; \frac{\Delta t}{2\Delta x}\,[\;\mathbf{F}^x_i + \mathbf{F}^x_{i+1} - \mathbf{F}^x_{i-1} - \mathbf{F}^x_i\;] + \Delta t\;\mathbf{S}_i\,. \tag{1.14}$$

If we cancel out terms we obtain

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n \;\; - \;\; \frac{\Delta t}{2\Delta x}\,[\;\mathbf{F}^x_{i+1} - \mathbf{F}^x_{i-1}\;] + \Delta t\;\mathbf{S}_i\,. \tag{1.15}$$

So, as we have seen, the choice (1.12) makes (1.10) fully equivalent to the standard, centered, second order accurate FD approach for first order derivatives. As it is well known, this choice is prone to developing high-frequency noise in presence of steep

gradients, like the ones appearing in black hole simulations. For this reason, artificial viscosity terms are usually required in order to suppress the spurious high-frequency modes [6].

We will consider here more general flux formulae, namely

$$F_{i+1/2} \;=\; f(u_L\,,\; u_R)\,, \tag{1.16}$$

where u_L, u_R stand for the left and right predictions for the dynamical field u at the chosen interface (arrows and dots, respectively, in Fig. 1.1). In the (piecewise) linear case, they are given by

$$u^L = u_i + 1/2\;\sigma_i\;\Delta x \qquad u^R = u_{i+1} - 1/2\;\sigma_{i+1}\;\Delta x\;, \tag{1.17}$$

where σ_i stands for the slope of the chosen field in the corresponding cell.

A sophisticated choice is provided by the 'shock-capturing' methods (see Ref. [4] for a review). These are methods based in Godunov's method. The idea, as we have seen, is to consider the jump at the interface as a physical one (not just a numerical artifact). The characteristic decomposition of (the principal part of) the system is then used in order to compute some physically sound interface Flux. These advanced methods have been common practice in Computational Fluid Dynamics since decades. They were adapted to the Numerical Relativity context nineteen years ago [7], for dealing with the spherically symmetric (1D) black-hole case. They are still currently used in Relativistic Hydrodynamics codes, but their use in 3D black hole simulations has been limited by the computational cost of performing the characteristic decomposition of the evolution system at every single interface.

More recently, much simpler alternatives have been proposed, which require just the knowledge of the characteristic speeds, not the full characteristic decomposition. Some of them have yet been implemented in Relativistic Hydrodynamics codes [8]. Maybe the simplest choice is the so called local Lax-Friedrichs (LLF) flux formula [9] or Rusanov formula [10]

$$f(u_L\,,\; u_R) = \frac{1}{2}\;[\; F(u_L) + F(u_R) + c\;(u_L - u_R)\;]\;, \tag{1.18}$$

where the coefficient c depends on the values of the characteristic speeds at the interface, namely:

$$c = max(\;\lambda_L\,,\lambda_R\;)\;, \tag{1.19}$$

where λ is the spectral radius (the absolute value of the biggest characteristic speed). We must point out that in this case, unlike most of non-centered finite volume methods, we will only need the biggest eigenvalue of the Jacobian matrix of the system which, in

our case, we can calculate analytically. We also want to stress out that we have have used the notation $F(u_L)$ and $F(u_R)$ to emphasize that we first reconstruct the fields at the interfaces and then we calculate the fluxes there.

If we take the LLF choice (1.18), combined with (1.17) and (1.15) we obtain a discretization such as

$$\begin{aligned}\mathbf{u}_i^{n+1} = \mathbf{u}_i^n \quad - \quad & \frac{\Delta t}{2\Delta x} \left[\mathbf{F}^x(u_{i+1} - \sigma_{i+1}\Delta x/2) - \mathbf{F}^x(u_{i-1} + \sigma_{i-1}\Delta x/2) \right. \\ & \left. -c \, (u_{i+1} + u_{i-1} - 2u_i - \sigma_{i+1}\Delta x/2 + + \sigma_{i-1}\Delta x/2) \right] \\ & + \Delta t \, \mathbf{S}_i \end{aligned} \tag{1.20}$$

And, if we forget about the reconstruction slopes for the sake of simplicity, we obtain

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n \quad - \quad \frac{\Delta t}{2\Delta x} \left[\mathbf{F}^x_{i+1} - \mathbf{F}^x_{i-1} - c \, (u_{i+1} + u_{i-1} - 2u_i) \right] + \Delta t \, \mathbf{S}_i \tag{1.21}$$

When compared to the centered FD discretization (1.15), we can see how the additional terms play the role of a numerical dissipation, because we can interpret that the equation that we are now resolving is

$$\frac{\partial u}{\partial t} + \frac{\partial F}{\partial x} - c \, \Delta x \frac{\partial^2 u}{\partial x^2} = S \ , \tag{1.22}$$

And it is well known that the second derivative terms play a dissipative or explosive role. In this case, because of the sign of the term, it is dissipative, therefore it improves the stability of the method. In this sense, a much more dissipative choice for (1.21) would be

$$c = \frac{\Delta x}{\Delta t} \ , \tag{1.23}$$

Which is in fact the most dissipative choice possible, because the Courant criteria does not allow to assign this dissipative term coefficient a higher value. This choice is equivalent to the Lax-Friedrichs method that we can find in [4]

$$\mathbf{u}_i^{n+1} = \frac{1}{2}(\mathbf{u}_{i+1}^n + \mathbf{u}_{i-1}^n) - \frac{\Delta t}{2\Delta x} \left[\mathbf{F}^x_{i+1} - \mathbf{F}^x_{i-1} \right] + \Delta t \, \mathbf{S}_i \tag{1.24}$$

1.3 Flux splitting approach

In the flux formulae approach (1.16), the information coming from both sides is processed at every interface, where different components are selected from either side in order to build up the flux there. We will consider here an alternative approach, in which the information is processed instead at the grid nodes, by selecting there the components of the flux that will propagate in either direction (flux splitting approach) [4].

The flux-splitting analogous of the original LLF formula (1.18, 1.19) can be obtained by splitting the flux into two simple components

$$F^{\pm}(u_i) = F(u_i^{\pm}) \pm \lambda_i \; u_i^{\pm} \,, \tag{1.25}$$

where λ will be again the spectral radius at the given grid point. Each component is then reconstructed separately, leading to one-sided predictions at the neighbour interfaces. The final interface flux will be computed then simply as

$$F_{i+1/2} \;=\; \frac{1}{2} \, (F_L^+ + F_R^-) \,. \tag{1.26}$$

This method can also be expressed as a modified LLF flux formula, namely

$$f(u_L \,,\; u_R) = \frac{1}{2} \, [\; F(u_L^+) + F(u_R^-) + \lambda_L \; u_L^+ - \lambda_R \; u_R^- \;] \,. \tag{1.27}$$

The main difference between the original LLF flux formula (1.18) and the flux-splitting variant (1.27) is that in the last case there is a clear-cut separation between the contributions coming from either the left or the right side of the interface, as it can clearly be seen in (1.26). In this way, one has a clear vision of the information flux in the numerical algorithm. The information from F^+ components propagates in the forward direction, whereas the one from F^- components propagates backwards. This simple splitting provides in this way some insight that can be useful for setting up suitable boundary conditions. Moreover, it opens the door to using different slopes for the reconstruction of each flux component. We will see below how to take advantage of this fact in order to improve space accuracy.

Third order accuracy

As it is well known, the use of a consistent piecewise-linear reconstruction results generically into a second-order space accuracy. A convenient choice is given by the centered

slope

$$\sigma^C = \frac{1}{2\Delta x}\,(u_{i+1} - u_{i-1}). \tag{1.28}$$

This is a good default choice (Fromm choice [4]), leading to reliable second-order accurate algorithms .

More general second-order algorithms can be obtained by replacing the centered slope σ^C by any convex average of the left and right slopes,

$$\sigma^L = (u_i - u_{i-1})/\Delta x\ , \qquad \sigma^R = (u_{i+1} - u_i)/\Delta x\ . \tag{1.29}$$

In some applications, however, second order accuracy is not enough. The leading (third order) error is of the dispersion type, affecting the numerical propagation speeds. In the FD approach, this can be improved by using a fourth-order-accurate algorithm in combination with a fourth-order artificial dissipation term (which constitutes itself the leading error term). The resulting combination is third-order accurate.

In the standard FV approach, the current way of getting (piecewise) third-order accuracy would be instead to replace the piecewise linear reconstruction by a piecewise parabolic one. The prototypical example is provided by the well known piecewise parabolic methods (PPM). The main complication of this strategy is that node values would no longer represent the cell averages of a given dynamical field. This would increase the complexity of the reconstruction process and the computational cost of the resulting algorithm.

There is a much simpler alternative, which takes advantage of the Flux splitting (1.25). The idea is to consider the resulting one-sided components $F^{\pm}$ as independent dynamical fields, each one with its own slope. The surprising result is that the choice

$$\sigma^+ = \frac{1}{3}\,\sigma^L + \frac{2}{3}\,\sigma^R\ , \qquad \sigma^- = \frac{2}{3}\,\sigma^L + \frac{1}{3}\,\sigma^R \tag{1.30}$$

leads, after the recombination (1.26), to a third-order accurate algorithm. The coefficients in (1.30) are unique: any other combination leads just to second-order accuracy.

Following the reasoning of Appendix A, we can prove the previous statements with a simple equation such as the advection equation

$$\frac{\partial u}{\partial t} + v\frac{\partial u}{\partial x} = \frac{\partial u}{\partial t} + \frac{\partial F}{\partial x} = 0 \tag{1.31}$$

Where we have assumed v is constant and therefore $F = v\ u$. The spatial part can be discretized as

$$\frac{\partial u_i}{\partial t} + \frac{F_{i+1/2} - F_{i-1/2}}{\Delta x} = 0 \tag{1.32}$$

If we use (1.27) and we keep in mind that now, with this flux splitting approach we can use different slopes we can say that

$$u_L^{\pm} = u_i + 1/2\ \sigma_i^{\pm}\ \Delta x \qquad u_R^{\pm} = u_{i+1} - 1/2\ \sigma_{i+1}^{\pm}\ \Delta x\ , \tag{1.33}$$

And we obtain

$$\begin{aligned} \frac{\partial u_i}{\partial t} \quad + \quad & \frac{(v+\lambda_i)(u_i + \sigma_i^+ \Delta x/2) + (v-\lambda_{i+1})(u_{i+1} - \sigma_{i+1}^- \Delta x/2)}{2\Delta x} \\ - \quad & \frac{(v+\lambda_{i-1})(u_{i-1} + \sigma_{i-1}^+ \Delta x/2) + (v-\lambda_i)(u_i - \sigma_i^- \Delta x/2)}{2\Delta x} = 0 \end{aligned} \tag{1.34}$$

If we now choose the plus and minus slopes as a linear combination of the left and right slopes

$$\sigma^+ = a\ \sigma^L + b\ \sigma^R\ , \qquad \sigma^- = c\ \sigma^L + d\ \sigma^R \tag{1.35}$$

And we use (1.29) in (1.34), we obtain

$$\begin{aligned} \frac{\partial u_i}{\partial t} \quad + \quad & \frac{v}{2\Delta x}\left[\frac{a}{2}u_{i-2} + \left(-a + \frac{b}{2} - \frac{c}{2} - 1\right) u_{i-1}\right] \\ + \quad & \frac{v}{2\Delta x}\left[\left(\frac{a}{2} - b + c - \frac{d}{2}\right) u_i + \left(\frac{b}{2} + 1 - \frac{c}{2} + d\right) u_{i+1} - \frac{d}{2}u_{i+2}\right] \\ + \quad & \frac{1}{2\Delta x}\left[\frac{a}{2}\lambda_{i-2}u_{i-2} + \left(-a + \frac{b}{2} + \frac{c}{2} - 1\right)\lambda_{i-1}u_{i-1} + \left(2 + \frac{a}{2} - b - c + \frac{d}{2}\right)\lambda_i u_i\right] \\ + \quad & \frac{1}{2\Delta x}\left[\left(\frac{b}{2} - 1 + \frac{c}{2} - d\right)\lambda_{i+1}u_{i+1} + \frac{d}{2}\lambda_{i+2}u_{i+2}\right] = 0 \end{aligned} \tag{1.36}$$

If we use now the Taylor series

$$
\begin{aligned}
u_{i+1} &= u_i + \Delta x\frac{\partial u}{\partial x} + \frac{\Delta x^2}{2}\frac{\partial^2 u}{\partial x^2} + \frac{\Delta x^3}{3!}\frac{\partial^3 u}{\partial x^3} + \frac{\Delta x^4}{4!}\frac{\partial^4 u}{\partial x^4} + O(\Delta x^5)\\
u_{i+2} &= u_i + 2\Delta x\frac{\partial u}{\partial x} + 2\Delta x^2\frac{\partial^2 u}{\partial x^2} + \frac{4\Delta x^3}{3}\frac{\partial^3 u}{\partial x^3} + \frac{2\Delta x^4}{3}\frac{\partial^4 u}{\partial x^4} + O(\Delta x^5)\\
u_{i-1} &= u_i - \Delta x\frac{\partial u}{\partial x} + \frac{\Delta x^2}{2}\frac{\partial^2 u}{\partial x^2} - \frac{\Delta x^3}{3!}\frac{\partial^3 u}{\partial x^3} + \frac{\Delta x^4}{4!}\frac{\partial^4 u}{\partial x^4} + O(\Delta x^5)\\
u_{i-2} &= u_i - 2\Delta x\frac{\partial u}{\partial x} + 2\Delta x^2\frac{\partial^2 u}{\partial x^2} - \frac{4\Delta x^3}{3}\frac{\partial^3 u}{\partial x^3} + \frac{2\Delta x^4}{3}\frac{\partial^4 u}{\partial x^4} + O(\Delta x^5)
\end{aligned}
\tag{1.37}
$$

in combination with (1.36), we obtain

$$
\begin{aligned}
&\frac{\partial u}{\partial t} + v\frac{\partial u}{\partial x} + (a+b-c-d)\,\frac{v\Delta x}{4}\frac{\partial^2 u}{\partial x^2} + (-3a-3d+2)\,\frac{v\Delta x^2}{12}\frac{\partial^3 u}{\partial x^3}\\
&\qquad + (7a+b-c-7d)\,\frac{v\Delta x^3}{48}\frac{\partial^4 u}{\partial x^4} + (a+b+c+d-2)\,\frac{\Delta x}{4}\frac{\partial^2 \lambda u}{\partial x^2}\\
&+ (-a+d)\,\frac{\Delta x^2}{4}\frac{\partial^3 \lambda u}{\partial x^3} + (7a+b+c+7d-2)\,\frac{\Delta x^3}{48}\frac{\partial^4 \lambda u}{\partial x^4} + O(\Delta x^4\frac{\partial^5 u}{\partial x^5})
\end{aligned}
\tag{1.38}
$$

We want to cancel the terms which are not originally in (1.31), except for the fourth derivative term with λ, as it will be the leading error of our third order method. From the second derivative terms we conclude that $a+b=c+d=1$. This is something we could have suspected from the very beginning, as this tells us that plus and minus slopes have to be weighted averages of left and right slopes. Cancellation of third derivative terms tell us that $a=d=1/3$ and, in combination with our result from second order derivatives, $b=c=2/3$, which is precisely our choice in (1.30). These coefficients lead us to

$$
\frac{\partial u}{\partial t} + v\frac{\partial u}{\partial x} = -\frac{\Delta x^3}{12}\frac{\partial^4 \lambda u}{\partial x^4} + O(\Delta x^4\frac{\partial^5 u}{\partial x^5})
\tag{1.39}
$$

We see how the dominant extra term which appears in our advection equation is a dissipative one, because it involves a fourth derivative. To understand why the sign of the coefficient makes it dissipative instead of explosive we can think of a sinusoidal function. The fourth derivative of a sinusoidal function is the same function, therefore we have a contribution which is exponentially decreasing with time. The fact that our leading error term is a fourth derivaive means that our method is a third order one, as we wanted to show.

1.4 The 1D Black Hole

As a first test, let us consider the Schwarzschild Black Hole in spherical coordinates. We will write the line element in the 'wormhole' form:

$$ds^2 = -(\ tanh\,\eta\ )^2\ dt^2 + 4m^2\ (\ cosh\,\eta/2\ )^4\ (\ d\eta^2 + d\Omega^2\ )\ , \tag{1.40}$$

which can be obtained from the isotropic form by the following coordinate transformation

$$r = m/2\ exp(\ \eta\ )\ . \tag{1.41}$$

The wormhole form (3.34) exploits the presence of a minimal surface (throat) at $\eta = 0$. It is manifestly invariant by the reflection isometry

$$\eta \leftrightarrow -\eta\ , \tag{1.42}$$

so that the numerical simulations can be restricted to positive values of η. The isometry (1.42) provides a very convenient boundary condition at the throat. Moreover (1.41) implies

$$dr = r\ d\eta \tag{1.43}$$

so that an evenly spaced grid in η corresponds to a geometrically increasing spacing in r. We can perform in this way long term simulations with a single grid of a limited size, as we will see below. This also allows to apply the standard boundary conditions in FV methods: two 'ghost' points are added by just copying the nearest neighbor values (or their time variation) for every dynamical field. The separation between incoming and outgoing information is automatically performed by the flux-splitting algorithm, so that boundary points are not special in this respect.

The simulations are performed with a spherically symmetric version of the Z3 formalism [11], as detailed in Appendix C. The free parameter n, governing the coupling with the energy constraint, is taken with unit value by default, but other similar values can be taken without affecting significatively the results, like $n = 4/3$, which corresponds to the CADM case [12]. Regarding gauge conditions, we are using the generalized harmonic prescription for the lapse [13]

$$(\partial_t - \mathcal{L}_\beta)\,\alpha = -f\ \alpha^2\ trK \tag{1.44}$$

with zero shift (normal coordinates). We take a constant (unit) value of the lapse as initial data. We can see in Fig. 1.2 the evolution of the lapse in a long-term simulation (up to $1000m$). We have chosen in this case $f = 2/\alpha$ (corresponding to the 1+log

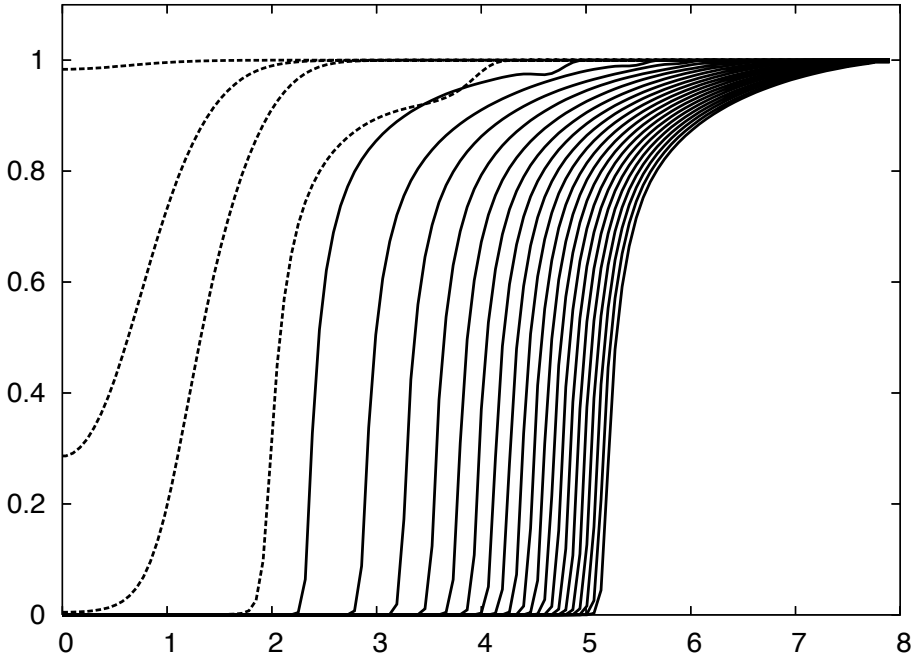


FIGURE 1.2: Long-term FV simulation of a 1D black hole, with a single mesh of 120 gridpoints. The evolution of the lapse is shown up to $1000m$, in intervals of $50m$ (solid lines). The dotted lines correspond to $1m$, $3m$, $5m$ and $25m$. Note that the plots tend to cumulate at the end, due to the exponential character of the grid, as given by (1.41). No slope limiters have been used in this simulation.

slicing), but similar results can be obtained with many other combinations of the form

$$f = a + b/\alpha \ , \tag{1.45}$$

where a and b are constant parameters.

Note that no slope limiters have been used in the simulation shown in Fig. 1.2. This can seem surprising at the first sight, but it can be better understood by having a look at the next chapter

As an accuracy check, we monitor the mass function [14], which is to be constant in space and time for the Schwarzschild case, independently of the coordinate system. In Fig. 1.3, we compare (the L_2 norm of) the errors in the mass function between a third-order FV simulation (without slope limiters) and the corresponding FD simulation (including a fourth order dissipation term like the one in ref. [15] with $\epsilon = 0.015$). We see that the FD method shows bigger errors at late times. One can argue that the leading error in the FD simulation is given by the dissipation terms, so that one can modify the result by lowering the numerical dissipation coefficient. However, lowering the viscosity coefficient used in Fig. 1.3, would result into a premature code crashing, like the one shown in the Figure for a strictly fourth order FD run, without the artificial dissipation term.

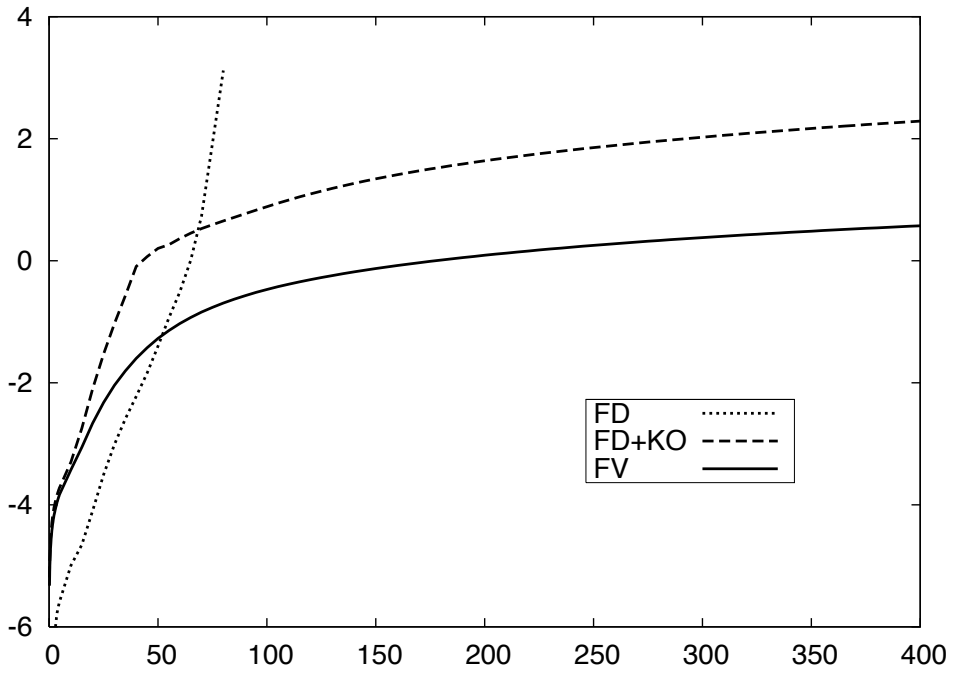


FIGURE 1.3: Time evolution of the error in the mass function (logarithm of the L_2 norm) for three different numerical algorithms. The strictly fourth-order FD method, without extra dissipation terms, is the most accurate as expected, but crashes after a short time (measured in units of m). The other two algorithms (third-order accurate) get similar errors at early times, but the FV one performs much better in the long term than the FD with standard Kreiss-Oliger dissipation. The dissipation coefficient has been taken as low as allowed by code stability (see the text). All simulations were obtained with a single mesh of 120 gridpoints and using the 1+log slicing prescription.

We can understand the need for dissipation by looking at the sharp collapse front in Fig. 1.2. We know that this is not a shock: it could be perfectly resolved by increasing the grid resolution as needed. In this way we can actually get long-term 1D black hole simulations, with a lifetime depending on the allowed resolution. This 'brute force' approach, however, can not be translated into the 3D case, where a more efficient management of the computational resources is required. This is where dissipation comes into play, either the numerical dissipation built in FV methods or the artificial one which is routinely added to fourth-order FD methods. Dissipation is very efficient in damping sharp features, corresponding to high-frequency Fourier modes. As a result, the collapse front gets smoothed out and can be resolved without allocating too many grid points. However, the more dissipation the more error. In this sense, Fig. 1.3 shows that adaptive viscosity built in the proposed FV method provides a good compromise between accuracy and computational efficiency.

Note that the error comparison is independent of the selected resolution. This is because the two stable methods in Fig. 1.3 are of third order accuracy, as confirmed by the local convergence test shown in Fig. 1.4 (solid line, corresponding to $t = 10\,m$). In the

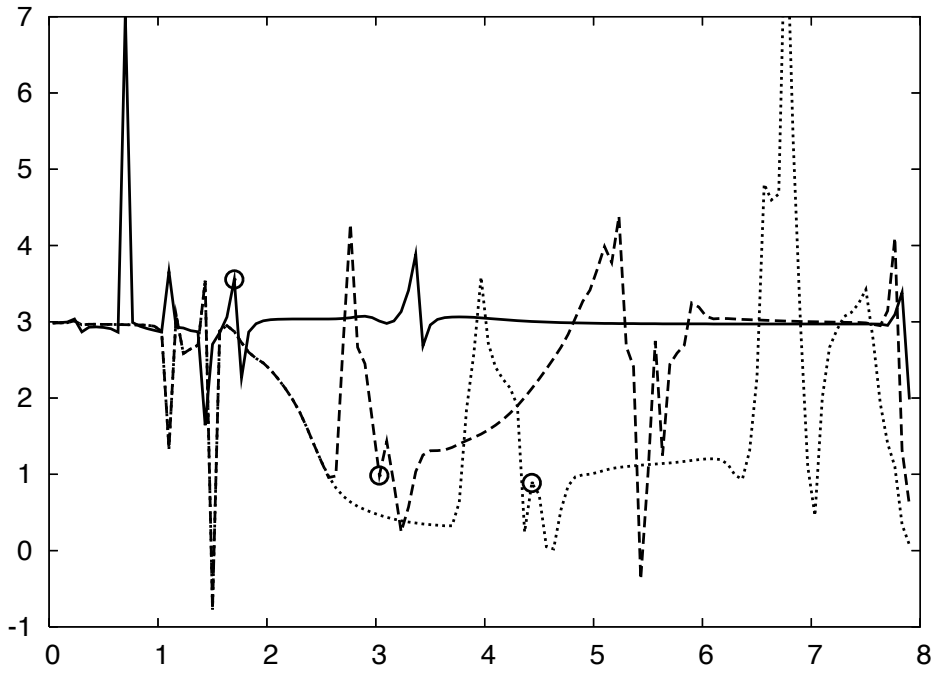


FIGURE 1.4: Local convergence evolution for the mass function in a 1D black hole simulation. We can see the predicted third-order accuracy, when using the proposed slopes (1.30), around $t = 10\,m$ (solid line). At $t = 100\,m$ (dashed line), we yet see the downgrade in the regions around the collapse front (the apparent horizon position is marked with a circle). As the collapse front propagates (dotted line, corresponding to $t = 400\,m$), we can see the growth of the affected regions, specially the one behind the front.

long term, however, large errors develop around the collapse front, downgrading the local convergence rate in the neighbor regions (dashed and dotted lines in Fig. 1.4, corresponding to $t = 100\,m$ and $t = 400\,m$, respectively). This can not be seen as a failure of the algorithm properties, but rather as consequence of large errors in a highly non-linear context. This also shows that in simulations oriented to compute gravitational wave patterns, the waveform extraction zone must be safely located, away both from the outer boundary and from the collapse front.

1.5 The 3D Black Hole

The 1D algorithm (1.10) can be easily adapted to the full three-dimensional (3D) case:

$$\begin{aligned} \mathbf{u}^{n+1}_{\{ijk\}} = \mathbf{u}^{n}_{\{ijk\}} \quad &- \quad \frac{\Delta t}{\Delta x} \left[\, \mathbf{F}^{x}_{\{i+1/2\, jk\}} - \mathbf{F}^{x}_{\{i-1/2\, jk\}} \, \right] \\ &- \quad \frac{\Delta t}{\Delta y} \left[\, \mathbf{F}^{y}_{\{i\, j+1/2\, k\}} - \mathbf{F}^{y}_{\{i\, j-1/2\, k\}} \, \right] \\ &- \quad \frac{\Delta t}{\Delta z} \left[\, \mathbf{F}^{z}_{\{ij\; k+1/2\}} - \mathbf{F}^{z}_{\{ij\; k-1/2\}} \, \right] \\ &+ \quad \Delta t \; \mathbf{S}_{\{ijk\}} \, . \end{aligned} \tag{1.46}$$

The structure of (1.46) suggests dealing with the 3D problem as a simple superposition of 1D problems along every single space direction. The stability analysis in Appendix A can then be extended in a straightforward way, showing that the strong stability requirement leads to a more restrictive upper bound on the timestep (in our case, using a cubic grid, this amounts to an extra 1/3 factor).

In cartesian-like coordinates, it is not so easy to take advantage of the reflection isometry (1.42). For this reason, we will evolve both the black-hole exterior and the interior domains. We can not use the η coordinate for this purpose, because the symmetry center would correspond to $\eta \to \infty$. We will take instead the initial space metric in isotropic coordinates, namely

$$\mathrm{d}l^2 = (1 + \frac{m}{2r})^4 \; \delta_{ij} \; dx^i dx^j \; . \tag{1.47}$$

We will replace then the vacuum black-hole interior by some singularity-free matter solution. To be more specific, we will allow the initial mass to have a radial dependence: $m = m(r)$ in the interior region. This allows to match a scalar field interior metric to (3.31) ('stuffed black-hole' approach [16]). The price to pay for using a regular metric inside the horizon is to evolve the matter content during the simulation: we have chosen the scalar field just for simplicity.

Let us consider initial data taken from a Schwarzschild black hole

$$ds^2 = -\alpha^2 dt^2 + (1 + \frac{M}{2r})^4 \; \delta_{ij} \; dx^i dx^j \, . \tag{1.48}$$

(isotropic coordinates). We will use the 'stuffed black hole' approach [16], by matching a scalar field interior metric to (1.48) (the scalar field will also evolve, see Appendix E for details). As gauge conditions we choose a singularity-avoidant slicing of the '1+log' type in normal coordinates (zero shift).

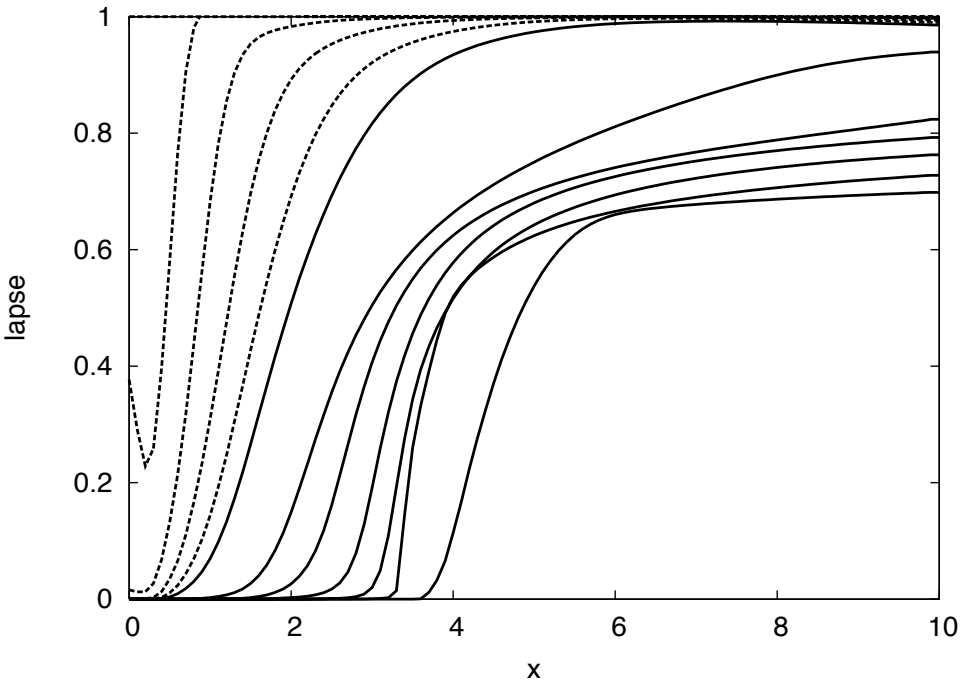


FIGURE 1.5: Lapse evolution in a 3D black hole simulation (zero shift). The dotted line profiles are plotted every 1M. The solid line ones are plotted every 5M, up to 35M, before boundary-related features become too important (the boundary is just at 10M).

We present in (Fig. 1.5) a low-resolution simulation ($\Delta x = 0.1M$) which proves the performance of our numerical method in 3D strong-field scenarios. Even in presence of steep gradients, the lapse profiles evolve smoothly.

References

[1] C. Bona and J. Massó, Phys. Rev. **D40** 1022 (1989).

[2] W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Veterling, *Numerical Recipes*, Cambridge University Press, (Cambridge 1989)

[3] S. Gottlieb, C.-W. Shu and E. Tadmor, SIAM Review **43**, 89 (2001).

[4] R. J. LeVeque, *Finite Volume Methods for Hyperbolic Problems*. Cambridge (Cambridge 2002).

[5] A. Arbona, C. Bona, J. Massó and J. Stela, Phys. Rev. **D60** 104014 (1999).

[6] B. Gustafson, H.O. Kreiss and J. Oliger, *Time dependent problems and difference methods*, Wiley, New York (1995).

[7] C. Bona and J. Massó, Phys. Rev. Lett. **D68** 1097 (1992).

[8] M. A. Aloy, J. A. Pons and J. M. Ibáñez, Computer Physics Commun. **120** 115 (1999).

[9] P. D. Lax, Comm. Pure Appl. Math., VII, pp159193. (1954)

[10] V. V. Rusanov, J. Comput. Mat. Phys. USSR **1** 267 (1961).

[11] C. Bona, T. Ledvinka and C. Palenzuela, Phys. Rev. **D66** 084013 (2002).

[12] C. Bona, T. Ledvinka, C. Palenzuela, M. Žáček, Phys. Rev. **D69** 064036 (2004).

[13] C. Bona, J. Massó, E. Seidel, and J. Stela, Phys. Rev. Lett. **75** 600 (1995).

[14] G. Lemaître, Rev. Mod. Phys. **21** 357 (1949).

[15] L. Baiotti and L. Rezzolla, Phys. Rev. Lett. **97** 141101 (2006).

[16] A. Arbona et al., Phys. Rev. **D57** 2397 (1998).

[17] O. A. Liskovets, Differential equations I 1308-1323 (1965).

Chapter 2

FDOC discretization algorithms

2.1 Introduction

In the previous chapter we devised a centered finite volume numerical algorithm which provides third order accuracy using a piecewise linear reconstruction. This space discretization scheme, together with a Runge-Kutta algorithm for the time discretization in the context of the MoL technique, which allows a clear separation between space and time discretizations, helped us to successfully perform the collapse of a single black hole in spherical coordinates using the Z3 formalism of the Einstein equations and allowed us to simulate a collapse of a 3D black hole further than ever before with the Z3 formalism.

We also showed a comparison between the accuracy of our method and the method used in [1] in terms of the error in the mass function. This is because regarding space accuracy, the most common approach in Numerical Relativity is to use a centered FD, nth-order accurate method (being n even), combined with some artificial dissipation term of the Kreiss-Oliger (KO) kind [2]. The dissipation applied has to be tuned with a single parameter and this may be a difficulty in some cases, where dealing with the black hole interior would require an amount of dissipation which can be instead too big for the exterior region (see for instance Ref. [1]), not to mention the fact that the optimal value of the parameter (i.e. the one that gets rid of the high frequency noise without losing much accuracy) changes from one kind of simulation to another.

However, just as we argued the efficiency of centered FV methods compared to high resolution shock capturing methods, we could argue here that a FD method with KO dissipation is way more efficient than our method in terms of computational cost. This is indeed true if we implement the method in a pedestrian way, that is, calculating and storing all the slopes and intermediate quantities. If we, on top of that, double the

number of points used in each direction to keep track of the values of the fields and fluxes at the grid points and at the interfaces, we end up with an impractical implementation of the method to perform 3D simulations of the Einstein equations. To avoid that, and taking advantage of the fact that we are not using slope limiters, we can repeat a process similar to the calculations that go from (1.18) to (1.21) and see what happens. In fact, if we depart from (1.36) and we remember that we were using an advection equation, where the fluxes are $F = v\ u$, and use our third order accurate result, with coefficients $a = d = 1/3$ and $b = c = 2/3$ we end up with the following equation

$$\begin{aligned} &\frac{\partial u_i}{\partial t} + \frac{1}{\Delta x}\left[\frac{1}{12}F_{i-2} - \frac{2}{3}F_{i-1} + \frac{2}{3}F_{i+1} - \frac{1}{12}F_{i+2}\right] \\ &+\frac{1}{\Delta x}\left[\frac{1}{12}\lambda_{i-2}u_{i-2} - \frac{1}{3}\lambda_{i-1}u_{i-1} + \frac{1}{2}\lambda_i u_i - \frac{1}{3}\lambda_{i+1}u_{i+1} + \frac{1}{12}\lambda_{i+2}u_{i+2}\right] = 0 \end{aligned} \tag{2.1}$$

With all these simplifications, the proposed centered FV method can be interpreted just as a fourth order centered FD method combined with a dissipation term (as we saw in the last chapter that these lambda terms have a dissipative role). In fact, this is an 'adaptive viscosity' generalization of the finite difference (FD) algorithms with KO dissipation, which look like

$$\begin{aligned} &\frac{\partial u_i}{\partial t} + \frac{1}{\Delta x}\left[\frac{1}{12}F_{i-2} - \frac{2}{3}F_{i-1} + \frac{2}{3}F_{i+1} - \frac{1}{12}F_{i+2}\right] \\ &+\frac{\sigma}{\Delta x}\left[\frac{1}{12}u_{i-2} - \frac{1}{3}u_{i-1} + \frac{1}{2}u_i - \frac{1}{3}u_{i+1} + \frac{1}{12}u_{i+2}\right] = 0 \end{aligned} \tag{2.2}$$

In the case of a centered fourth order FD method with a fourth order dissipative term of the KO kind, which results, as in our case, in a third order accurate method. In this case σ represents the arbitrarily tuned dissipation parameter whereas in (2.1), the values of the dissipation coefficients are entirely prescribed by the numerical algorithms: there are no arbitrary parameters, unlike the KO case.

We see that the lack of slope limiters has allowed to implement the method presented in the previous chapter as a FD method with some artificial viscosity term, therefore matching the efficiency of the FD+KO methods.

Nowadays, current trend in the numerical relativity community is to use a FD algorithm of nth order and to apply a KO dissipation of n+2th-order on top of that, so that the leading error in the solution is a dispersive one, of order n+1. Choosing a dissipation of order n+2 implies that the stencil, the number of points required to evaluate each

derivative at every point, will be increased by two units (one at each side as the method is centered) with respect to the stencil used by the FD method. In the end, the number of points used is the same as if a centered FD n+2th-order method was used. So, if one uses a FD difference algorithm of order n+2 and a dissipation term of the same order, ends up with a n+1th-order accurate method for the same price. The leading error is now precisely the dissipation applied, which one can tune with a single parameter. Our point is that centered Finite Volume methods can provide alternative n+1th-order accurate algorithms in which the built-in dissipation is automatically adapted to the requirements of either the interior or exterior black hole regions.

Not only we have shown that this alternative can match the current trend in terms of computational efficiency but, if we have to draw conclusions from the results presented in last chapter, it seems like, when using this centered FV method high frequency noise is absent even at accuracy levels that one cannot reach with FD+KO without having such noise. Does this mean that it is impossible for our method to show this behaviour or is it only well behaved under certain conditions? To answer this question we must review some background theory in next sections.

2.2 Total Variation

The study of hyperbolic conservation laws, represented by

$$\partial_t u + \partial_x f(u) = 0 \ , \tag{2.3}$$

is a classical topic in Computational Fluid Dynamics (CFD). We have noted here by u a generic array of dynamical fields, and we will assume strong hyperbolicity, so that the characteristic matrix

$$A(u) = \partial f / \partial u \tag{2.4}$$

has real eigenvalues and a full set of eigenvectors.

As it is well known, the system (2.3) admits weak solutions, so that the components of u may show piecewise-smooth profiles. Standard finite-difference schemes, like the Lax-Wendroff [3] or MacCormack [4] ones, produce spurious overshots and oscillations at non-smooth points which can mask the physical solutions, even leading to code crashing. These deviations do not diminish with resolution, in analogy with the Gibbs phenomenon found in the Fourier series development of discontinuous functions.

This difficulty was overcome in the pioneering work of Godunov [5]. On a uniform computational grid $x_j = j \triangle x$, equation (2.3) can be approximated by the semi-discrete

equation

$$\partial_t u_j = -\frac{1}{\Delta x}\left(h_{j+1/2} - h_{j-1/2}\right) , \tag{2.5}$$

where the interface flux $h_{j+1/2}$ is computed by an upwind-biased formula from the neighbor grid nodes. In the scalar case, one can define the total variation of a discrete function as

$$TV(u) = \sum_j \left| u_j - u_{j-1} \right| . \tag{2.6}$$

In the case of systems, the total variation is defined as the sum of the total variation of the components. Godunov scheme is total-variation-diminishing (TVD), meaning that $TV(u)$ does not increase during numerical evolution. It is obvious that TVD schemes can not develop spurious oscillations: monotonic initial data preserve their monotonicity during time evolution. Moreover, the TVD property can be seen as a strong form of stability: any blow-up of the numerical solution is excluded, as far as it would increase the total variation.

Godunov scheme is the prototype of the so-called upwind-biased schemes, which require either the exact or some approximate form of spectral decomposition of the characteristic matrix (2.4). This makes them both computationally expensive and difficult to extend to the multidimensional case, as we have already argued. A much simpler alternative is provided by the local Lax-Friedrichs (LLF) scheme or Rusanov scheme [6] which we can recall from last chapter

$$h_{j+1/2} = \frac{1}{2}\left[f_{j+1} + f_j - \lambda_{j+1/2}\left(u_{j+1} - u_j\right) \right] , \tag{2.7}$$

where λ is the spectral radius of the characteristic matrix and we have taken

$$\lambda_{j+1/2} = max(\lambda_j, \lambda_{j+1}) . \tag{2.8}$$

It is clear from (2.5, 2.7) that the LLF scheme, like the Godunov one, is only first-order accurate in space (we are using piecewise constant reconstruction here). Second-order accuracy can be obtained following the Harten modified-flux approach [7], which was soon extended to very-high accuracy (up to 15th order) by Osher and Chakrabarthy [8]. The basic idea is to replace the lower order TVD flux $h_{j+1/2}$ by a modified flux $f_{j+1/2}$, obtained by some interpolation procedure involving a higher number of nodes.

All these high-resolution schemes require some form of flux-correction limiters in order to ensure the TVD property. As a consequence, accuracy is reduced to (at most) first order at non-sonic critical points, where the limiters come into play. In order to circumvent this problem, one can relax the TVD condition, demanding instead that the total variation

is bounded, that is

$$TV(u) \leq B \ , \tag{2.9}$$

where the upper bound B is independent of the resolution, but could depend on the elapsed time. Even if we are ready to relax the stronger TVD requirement, keeping the bound (2.9) is important from the theoretical point of view.

An interesting example of such TVB schemes was given by Shu [9], by softening the flux limiters proposed in Ref. [8]. Although the TVB property is proven for the schemes presented in [9], based on a linear flux-modification procedure, a rigorous proof is still unavailable for more complex cases. An important example is provided by the essentially-non-oscillatory (ENO) methods [10] [11], where the TVD property is relaxed in a different way. Numerical evidence shows that ENO schemes, as well as their weighted-ENO variants [12]-[14], deserve their name: the TVB property is satisfied in practice, even with time-independent bounds. An implementation of these high-resolution methods for the LLF Flux is given in Refs. [15] [16].

2.3 The Osher-Chakravarthy β-schemes

Following Ref. [8], let us consider the centered $2m-1$ order schemes:

$$\partial_t u_j = -C^{2m} f_j + (-1)^{m-1} \beta (\Delta x)^{2m-2} D_+^m D_-^{m-1} (df^+_{j-1/2} - df^-_{j-1/2}) \ , \tag{2.10}$$

where C^{2m} is the central $2m$th-order-accurate difference operator with a stencil of $2m+1$ grid points, and we have used the standard notation $D_\pm$ for the elementary difference operators. The flux differences $df^\pm$ are defined as follows:

$$df^+_{j+1/2} = f_{j+1} - h_{j+1/2} \qquad df^-_{j+1/2} = h_{j+1/2} - f_j \ , \tag{2.11}$$

where $h_{j+1/2}$ is any (lowest-order) TVD flux. The β parameter in the dissipative term in (2.10) is assumed to be positive: a necessary condition for stability.

The algorithms (2.10) can be put into an explicit flux-conservative form by replacing the lowest order flux $h_{j+1/2}$ in (2.5) by [8] [9]

$$f_{j+1/2} = h_{j+1/2} + \sum_{k=-m+1}^{m-1} \left(c^m_k \ df^-_{j+k+1/2} + d^m_k \ df^+_{j+k+1/2} \right) \ , \tag{2.12}$$

where

$$d^m_k = \nu^m_k - (-1)^k \beta \begin{pmatrix} 2m-2 \\ k+m-1 \end{pmatrix} \ , \qquad c^m_k = -d^m_{-k} \tag{2.13}$$

(we use here a compact notation), and

$$\nu_0^m = 1/2\ , \qquad \nu_k^m = -\nu^m{}_{-k} \qquad (k \neq 0) \tag{2.14}$$

$$\nu_{m-1}^m = (-1)^{m-1} \left[m \begin{pmatrix} 2m \\ m \end{pmatrix} \right]^{-1} \qquad (m > 1) \tag{2.15}$$

$$\nu_k^{m+1} = \nu_k^m + (-1)^k \frac{k}{m} \begin{pmatrix} 2m \\ m-k \end{pmatrix} \left[(m+1) \begin{pmatrix} 2m+2 \\ m+1 \end{pmatrix} \right]^{-1} . \tag{2.16}$$

The TVD property is enforced by limiting the flux differences $df^{\pm}$ (see [8], [9] for the details). As a generic example, $df^+_{j+k+1/2}$ in (2.12) is replaced by

$$minmod(\ df^+_{j+k+1/2}, bdf^+_{j+1/2}, bdf^+_{j-1/2}\ ), \tag{2.17}$$

where b is a compression factor. This replacement introduces a non-linear component in the linear flux-correction formula (2.12). The resulting scheme will be TVD if and only if:

$$C_{j+1/2} \equiv 1 + \sum_{k=-m+1}^{m-1} c_k{}^m \frac{df^-_{j+k+1/2} - df^-_{j+k-1/2}}{df^-_{j+1/2}} \geq 0 \tag{2.18}$$

$$D_{j-1/2} \equiv 1 + \sum_{k=-m+1}^{m-1} d_k{}^m \frac{df^+_{j+k+1/2} - df^+_{j+k-1/2}}{df^+_{j-1/2}} \geq 0\ . \tag{2.19}$$

$$\lambda_j \frac{\Delta t}{\Delta x} (C_{j+1/2} + D_{j+1/2}) \leq 1\ , \tag{2.20}$$

where we have assumed a time discretization based on the forward Euler step, so that the last condition provides an upper bound on the time step Δt.

2.4 Compression factor optimization

In the original paper [8], the ansatz

$$\beta \leq [\ m \begin{pmatrix} 2m \\ m \end{pmatrix}\ ]^{-1} \tag{2.21}$$

was used for getting a sufficient condition from (2.18, 2.19), amounting to a simple constraint on the range of the compression parameter b

$$0 < b \leq [\ 1 + 2\beta \begin{pmatrix} 2m-2 \\ m-1 \end{pmatrix}\ ]\ [\ \sum_{j=2}^{m} \frac{1}{2j-1}\ ]^{-1}\ . \tag{2.22}$$

Allowing for (2.22), the upper bound b_{max} increases with β, which is in turn bounded by (2.21). For the third-order scheme ($m = 2$), the optimal choice would then be $\beta = 1/12$, so that the compression parameter may reach $b_{max} = 4$, still preserving the TVD property. This means that, for monotonic profiles, the flux-correction limiters would act only where the higher order corrections in neighboring computational cells differ at least by a factor of four. This is not to be expected in practical, good resolution, simulations of smooth profiles, even when large gradients appear, which is precisely the case of numerical relativity simulations. This high-compression-factor property can be at the origin of the robust behavior of these schemes, even in their unlimited form, as we will see in the numerical applications presented below.

As far as we are proposing to use the unlimited version, it makes sense to find the choices of β that maximize the compression factor, going beyond the ansatz (2.21). Higher values of b_{max} can be actually obtained by a detailed case-by-case study of the original TVD conditions (2.18, 2.19). For instance, by reordering the terms in (2.19) we get

$$D_{j-1/2} \equiv 1 + \sum_{k=-m+1}^{m} (d_{k-1}^{\ m} - d_k^{\ m}) \, \frac{df^+_{j+k-1/2}}{df^+_{j-1/2}} \ \geq \ 0 \ , \tag{2.23}$$

where we assume $d_k^{\ m} = 0$ when $|k| \geq m$. A sufficient condition for (2.23) to hold is

$$1 + d_{-1}^{\ m} - d_0^{\ m} + b \sum_{k \neq 0} min(d_{k-1}^{\ m} - d_k^{\ m}, 0) \ \geq \ 0 \ , \tag{2.24}$$

which actually refines the former condition (2.22). The same reasoning shows that, allowing for (2.13), a sufficient condition for (2.20) to hold is:

$$\lambda_j \, \frac{\Delta t}{\Delta x} \, [\, d_{-1}^{\ m} - d_0^{\ m} + b \sum_{k \neq 0} max(d_{k-1}^{\ m} - d_k^{\ m}, 0) \,] \ \leq \ 1/2 \ . \tag{2.25}$$

For the simpler non-trivial cases we have (decreasing k order):

$$d_k^{\ 2} = (\ \beta - \frac{1}{12}, \quad \tfrac{1}{2} - 2\beta, \quad \beta + \frac{1}{12} \ ) \tag{2.26}$$

$$d_k^{\ 3} = (\ \frac{1}{60} - \beta, \ 4\beta - \frac{7}{60}, \quad \tfrac{1}{2} - 6\beta, \quad 4\beta + \frac{7}{60}, \ \ -\frac{1}{60} - \beta \ ) \ . \tag{2.27}$$

For $m = 2$, condition (2.29) leads then to:

$$1 + d_{-1}^{\ 2} - d_0^{\ 2} + b \ \ min(d_1^{\ 2}, 0) + b \ \ min(d_0^{\ 2} - d_1^{\ 2}, 0) + b \ \ min(-d_{-1}^{\ 2}, 0) \ \geq \ 0 \ , \tag{2.28}$$

Where we have assumed again that $d_k^m = 0$ when $|k| \geq m$. If we now use our calculations for ${d_k}^2$ we obtain:

$$\frac{7}{12} + 3\beta + b \;\; min(\beta - \frac{1}{12}, 0) + b \;\; min(\frac{7}{12} - 3\beta, 0) + b \;\; min(-\beta - \frac{1}{12}, 0) \;\; \geq \;\; 0 \;, \quad (2.29)$$

Which has the solutions:

$$b \quad \leq \quad 7/2 + 18\beta \qquad (\beta \leq \frac{1}{12}) \quad (2.30)$$

$$b \quad \leq \quad \frac{7 + 36\beta}{1 + 12\beta} \qquad (\frac{1}{12} \leq \beta \leq \frac{7}{36}) \;. \quad (2.31)$$

It follows that the optimal values for the third-order scheme are

$$\beta = \frac{1}{12} \;, \qquad b_{max} = 5 \;. \quad (2.32)$$

Notice that the value of b_{max} is now 5 instead of 4, which was the one obtained from the original ansatz (2.21). For the fifth-order scheme ($m = 3$), condition (2.29) leads instead to:

$$b \quad \leq \quad \frac{37 + 600\beta}{16} \qquad (\beta \leq \frac{1}{60}) \quad (2.33)$$

$$b \quad \leq \quad \frac{37 + 600\beta}{15 + 60\beta} \qquad (\frac{1}{60} \leq \beta \leq \frac{2}{75}) \quad (2.34)$$

$$b \quad \leq \quad \frac{37 + 600\beta}{7 + 360\beta} \qquad (\frac{2}{75} \leq \beta \leq \frac{37}{600}) \;. \quad (2.35)$$

It follows that the optimal values for the fifth-order scheme are

$$\beta = \frac{2}{75} \;, \qquad b_{max} = \frac{265}{83} \;. \quad (2.36)$$

Note that the ansatz (2.21) gives a smaller compression factor $b_{max} = 9/4$ and, more important, the optimal β value in this case is beyond the original bound $1/60$. Note also that the values of the compression parameter tend to diminish with the accuracy order of the algorithm. This suggests that higher-order cases $m > 3$ may not be so useful in the unlimited case.

2.5 Finite difference version

The linear flux-modification scheme described in the preceding section can be applied to any lower-order TVD flux. The case of the LLF flux (2.7) has actually been considered in [9]. Our objective here is to obtain a scheme which can be cast as a simple finite-difference algorithm, so that we will take advantage of the simplicity of the LLF flux (2.7), which can be written in flux-vector-splitting (FVS) form as we did in the past chapter:

$$h_{j+1/2} = f_j^+ + f_{j+1}^- \ , \qquad f_j^\pm \equiv \frac{1}{2}\,[f_j \pm \lambda_{j\pm1/2}\,u_j] \ . \tag{2.37}$$

The FVS form (2.37), like the original one (2.7), is just first-order accurate. We will extend it to higher-order accuracy by means of the Osher-Chakrabarthy algorithm, as described in the previous sections. The flux differences (2.11) in this case get the simple form:

$$df_{j+1/2}^\pm = 1/2\ [\, f_{j+1} - f_j \ \pm\ \lambda_{j+1/2}\,(u_{j+1} - u_j)\,] \ . \tag{2.38}$$

The linear character of this formula allows to get a compact finite-difference expression for the whole scheme. Allowing for (2.38), the semi-discrete algorithm (2.10) can be written as

$$\partial_t u_j = -C^{2m} f_j + (-1)^{m-1}\beta(\Delta x)^{2m-1} D_+^m D_-^{m-1}(\lambda_{j-1/2}\,D_- u_j) \ , \tag{2.39}$$

which amounts to assume a $2m$th-order-accurate central difference operator acting on the flux terms plus a dissipation operator of order $2m$ depending on the spectral radius λ. As we will see below, the resulting finite-difference scheme (2.39) provides a cost-effective alternative for CFD simulations.

Let us remark here that the choices (2.32, 2.36) derived in the previous section are optimal for a generic choice of the lowest-order TVD Flux. In the LLF case (2.7), however, it is clear that the spectral radius can be multiplied by a global magnifying factor $K > 1$, while keeping the TVD properties. Allowing for the finite-difference form (2.39) of the unlimited version, magnifying λ amounts to magnify β, that is:

$$(\beta,\ K\lambda) \qquad \Leftrightarrow \qquad (K\beta,\ \lambda) \ . \tag{2.40}$$

It follows that the values of the compression factor b_{max} obtained in the previous section must be interpreted just as lower-bound estimates. In particular, the equivalence (2.40) implies that any compression factor bound obtained for a particular value β_0 applies as well to all values $\beta > \beta_0$. This agrees with the interpretation of the second term in (2.39) as modelling numerical dissipation. On the other side, this dissipation term is actually

introducing the main truncation error. We will use then in what follows the β values in (2.32, 2.36), which are still optimal in the sense that they provide the lower numerical error compatible with the highest lower-bound for the compression parameter.

In the $m = 2$ case we obtain the following third order method from (2.39)

$$\begin{aligned}\frac{\partial u_i}{\partial t} + \frac{1}{\Delta x}\left[\frac{1}{12}F_{i-2} - \frac{2}{3}F_{i-1} + \frac{2}{3}F_{i+1} - \frac{1}{12}F_{i+2}\right] \\ + \frac{1}{\Delta x}\left[\frac{1}{12}\lambda_{i+3/2}(u_{i+2} - u_{i+1}) - \frac{1}{4}\lambda_{i+1/2}(u_{i+1} - u_i)\right] \\ + \frac{1}{\Delta x}\left[\frac{1}{4}\lambda_{i-1/2}(u_i - u_{i-1}) - \frac{1}{12}\lambda_{i-3/2}(u_{i-1} - u_{i-2})\right] = 0\end{aligned} \tag{2.41}$$

which is very similar to the third order method we devised in (2.1). In fact, if we take (1.27) instead of (2.38) we obtain precisely (2.1). This means that, with a slight modification, we end up with a method with enough theoretical support to justify the absence of slope limiters and which is also extendable to higher orders of accuracy. So for example in the m=3 case we get

$$\begin{aligned}\frac{\partial u_i}{\partial t} + \frac{1}{\Delta x}\left[-\frac{1}{60}F_{i-3} + \frac{3}{20}F_{i-2} - \frac{3}{4}F_{i-1} + \frac{3}{4}F_{i+1} - \frac{3}{20}F_{i+2} + \frac{1}{60}F_{i+3}\right] \\ + \frac{1}{\Delta x}\left[\frac{2}{75}\lambda_{i+5/2}(u_{i+2} - u_{i+3}) - \frac{2}{15}\lambda_{i+3/2}(u_{i+1} - u_{i+2})\right] \\ + \frac{1}{\Delta x}\left[\frac{4}{15}\lambda_{i+1/2}(u_i - u_{i+1}) - \frac{4}{15}\lambda_{i-1/2}(u_{i-1} - u_i)\right] \\ + \frac{1}{\Delta x}\left[\frac{2}{15}\lambda_{i-3/2}(u_{i-2} - u_{i-1}) - \frac{2}{75}\lambda_{i-5/2}(u_{i-3} - u_{i-2})\right] = 0\end{aligned} \tag{2.42}$$

which is a fifth order accurate method. We will call this family of methods Finite Difference Osher-Chakrabarthy (FDOC) from now on. Note that in the previous section we have refined the compression factor bounds given in the original paper [8] assuming that we will use these methods without slope limiters and with the efficient implementation we have presented in this section. We will perform now a battery of standard tests in one space dimension, covering advection, Burgers and Euler equations, in the following sections in order to show that the TVB property is fulfilled in practice for the selected values of the β parameter. We are not able, however, of getting the right result for compound shocks, arising from non-convex fluxes; this is illustrated by the Buckley-Leverett test simulations. This is beacuse the LLF flux formula (2.7) must be generalised for

non-convex fluxes. We will also consider some multidimensional tests cases with the Euler and magneto-hydrodynamics (MHD) equations, including the double Mach reflection and the Orszag-Tang 2D vortex problem. Total-variation-bounded behavior is evident in all the proposed cases, even with time-independent upper bounds.

2.5.1 Advection equation

Let us start by the scalar advection equation. This is the simplest linear case, but it allows to test the propagation of arbitrary initial profiles, containing jump discontinuities and corner points, departing from smoothness in many different ways. This is the case of the Balsara-Shu profile [12], which will be evolved with periodic boundary conditions.

We compare in Fig. 2.1 the numerical result with the exact solution after a single round trip, for two different resolutions. The third-order five-points formula from the proposed class (2.39) has been used with $\beta = 1/12$ in both cases or, in other words, FDOC3 (2.41) has been used in both cases. The propagation speed in the simulation agrees with the exact one, as expected for a third-order-accurate algorithm. The smooth regions are described correctly: even the height of the two regular maxima is not reduced too much

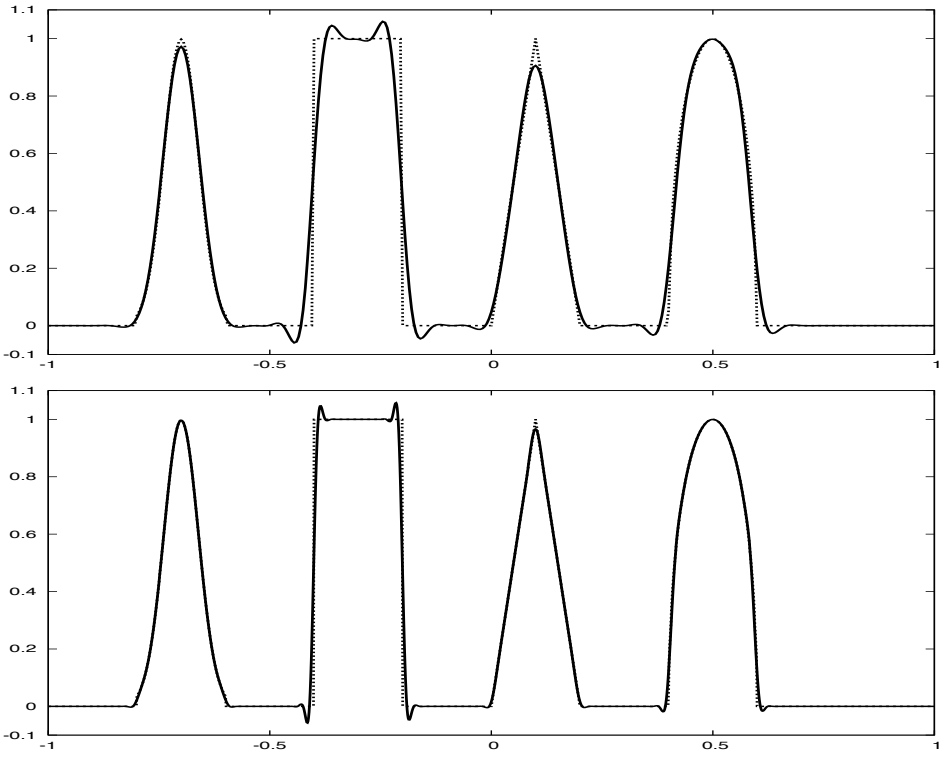


FIGURE 2.1: Advection of the Balsara-Shu profile in a numerical mesh of either 400 points (upper panel) or 800 points (lower panel). A third-order scheme ($m = 2$, $\beta = 1/12$) is used in both cases. The results are compared with the initial profile (dotted line) after a single round-trip.

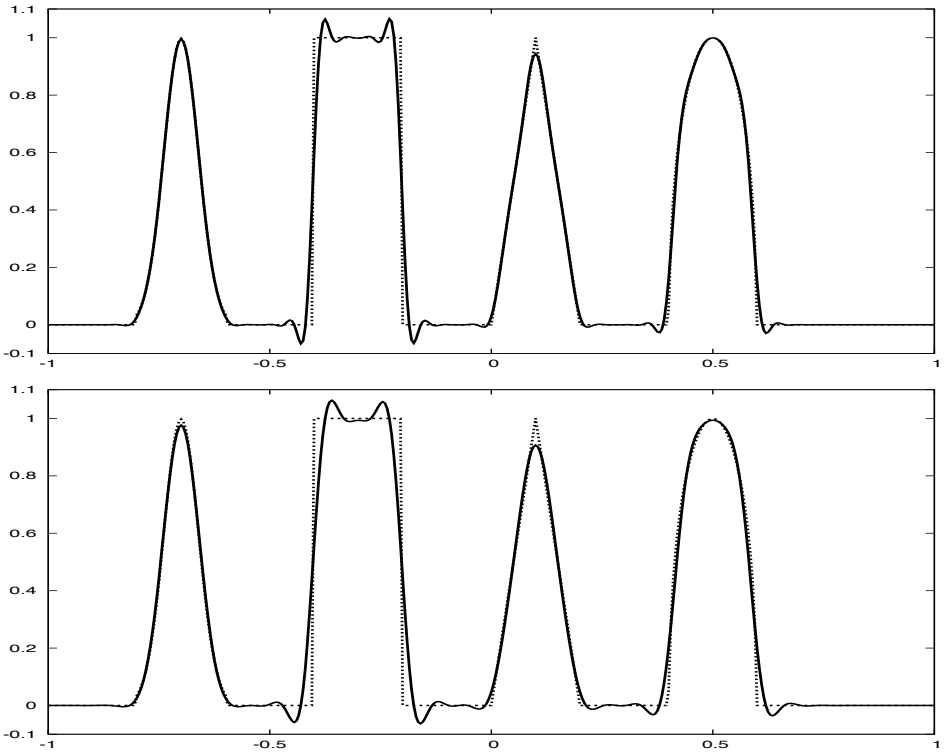


FIGURE 2.2: Same as in Fig. 2.1, but using a fifth-order scheme ($m = 3$) with $\beta = 2/75$ (upper panel). In the lower panel we show the results after ten round-trips. The same settings are used in both cases.

by dissipation, as expected for an unlimited algorithm with just fourth-order dissipation. There is a slight smearing of the jump slopes, as usual for contact discontinuities, which gets smaller with higher resolution.

Concerning monotonicity, it is clear that the total variation of the initial profile has increased by the riddles besides the corner points and, more visibly, near the jump discontinuities. By comparing the two resolutions, we see that the height of the overshots does not change. This means that, as in the case of the Gibbs phenomenon, there is no convergence by the maximum norm, although convergence by the L_2 or similar norms is apparent from the results. On the other hand, it is clear that the total variation is bounded for this fixed time, independently of the space resolution or, equivalently, the time step size. This is precisely the requirement for TVB.

We show in Fig. 2.2 the same simulation, in a 400 points mesh, for the fifth-order method ($m = 3$, $\beta = 2/75$, FDOC5 (2.42)). In the upper panel, corresponding to a single round-trip, we can see that one additional riddle appears at every side of the critical points, due to the larger (seven point) stencil. We show also in the lower panel the results of the same simulation after ten round-trips. The cumulative effect of numerical dissipation is clearly visible: the extra riddles tend to diminish. The total variation is not higher than

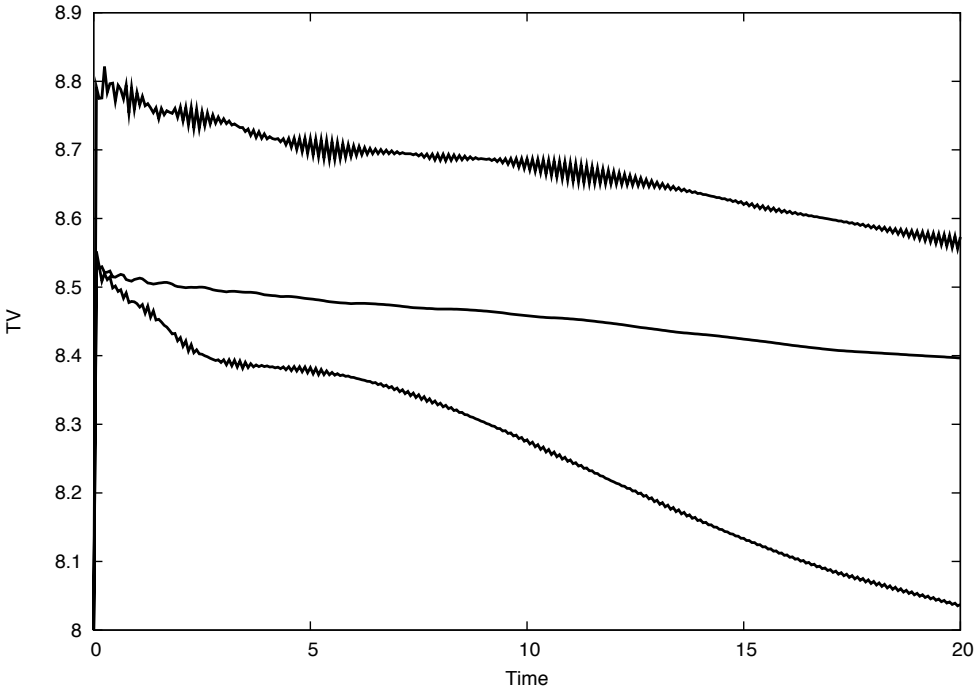


FIGURE 2.3: Advection equation. Time evolution of the total variation. The horizontal axis corresponds to the exact solution: $TV(u) = 8$. From top to bottom: FDOC5 scheme with 400 points, FDOC3 scheme with 800 points, and FDOC3 scheme with 400 points. After the initial increase, which depends on the selected method, the TV tends to diminish. Increasing resolution just reduces the TV diminishing rate.

the one after a single round trip. This statement can be verified by plotting, as we do in Fig. 2.3, the time evolution of $TV(u)$ for the different cases considered here. In all cases, a sudden initial increase is followed by a clear diminishing pattern. These numerical results indicate that the bound on the total variation is actually time-independent, beyond the weaker TVB requirement.

2.5.2 Burgers equation

Burgers equation provides a simple example of a genuinely non-linear scalar equation. A true shock develops from smooth initial data. We will compute here the evolution of an initial sinus profile, with fixed boundary conditions. We plot in Fig. 2.4 the numerical solution values versus (the principal branch of) the exact solution, at the time where the shock has fully developed. We compare 100 points with 200 points resolution (left and right panels, respectively), and also the 3rd-order and 5th-order schemes described previously (upper and lower panels, respectively). Concerning the resolution effect, we can see here again that the spurious oscillations affect mainly the points directly connected with the shock, in a number depending on the stencil size but independent of the resolution.

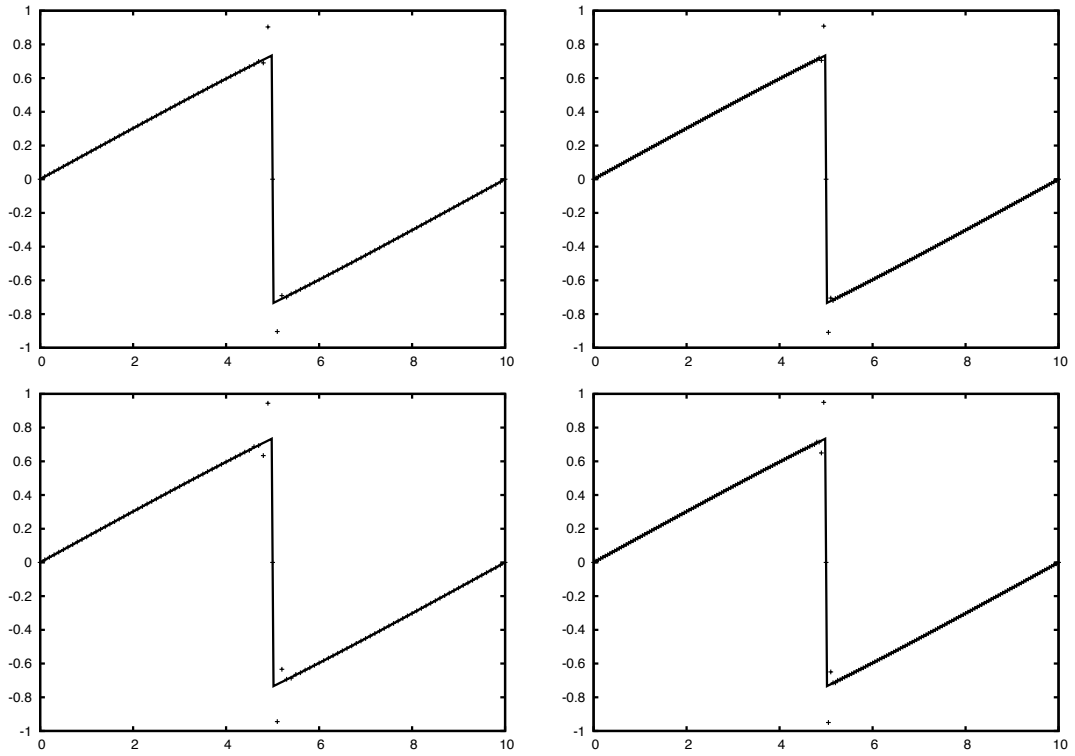


FIGURE 2.4: Burgers equation: evolution of an initial sinus profile. The numerical solution (point values) is plotted versus the exact solution (continuous line), for 100 points and 200 points resolution (left and right panels, respectively) and for the FDOC3 and the FDOC5 schemes (upper and lower panels, respectively).

These conclusions are fully confirmed by a second simulation, obtained by adding a constant term to the previous initial profile, that is

$$u(x) = \frac{1}{2} + sin(\frac{x\,\pi}{5})\,, \tag{2.43}$$

with periodic boundary conditions. We can see in Fig. 2.5 that a shock again develops, but it does no longer stand fixed: it propagates to the right. Note that the plot shown corresponds to $t = 7$. We can confirm in this case that both the number of spurious riddles and the magnitude of the overshots do not increase with resolution, although it is larger in this case than in the static shock one. We can confirm also that these effects increase with the order-of-accuracy of the scheme: the larger stencil adds one more riddle at every side and slightly larger overshots.

These results clearly indicate convergence in the L_1 or similar norms (but of course not in the maximum norm). Let us actually perform a convergence test by considering the initial profile [17]

$$u(x,0) = 1 + \frac{1}{2}\,sin(\pi\,x)\,, \tag{2.44}$$

which is smooth up to $t = 2/\pi$.

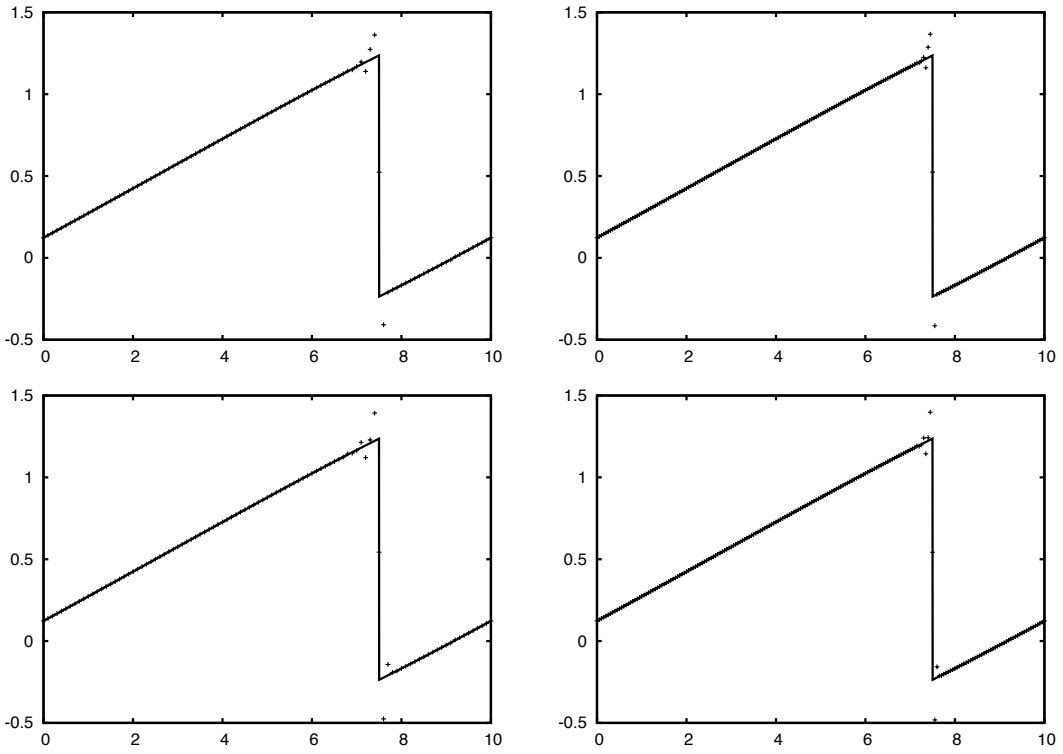


FIGURE 2.5: Same as in the previous figure, but now for a moving sinus profile. The numerical solution (point values) is plotted versus the exact solution (continuous line), for 100 points and 200 points resolution (left and right panels, respectively) and for the FDOC3 and the FDOC5 schemes (upper and lower panels, respectively).

We show in Table 2.1 the errors at time $t = 0.3$, where the shock has not yet appeared. The first group of values corresponds to the third-order method, and this is confirmed by the data both in the L_1 and the L_∞ norms. The second group of values corresponds to the fifth-order method, but only third-order accuracy is obtained from the numerical values. This is because we keep using the third-order Runge-Kutta algorithm (B.3) for the time evolution. In order to properly check the space discretization accuracy, we include a third group of values, obtained with the same algorithm, but with a much smaller time step in order to lower the time discretization error: the leading error term is then due to the space discretization and the expected fifth order accuracy is confirmed by the numerical results, although the L_∞ norm shows a slightly decreasing convergence rate for the higher resolution results.

2.5.3 Buckley-Leverett problem

A more demanding test, still for the scalar case, is provided by the Buckley-Leverett equation which models two-phase flows that arise in oil-recovery problems [18]. This

Nx	L_1 error	L_1 order	L_∞ error	L_∞ order
160	7.22579 E-6	2.998	5.17334 E-5	2.981
320	9.04719 E-7	2.999	6.55306 E-6	2.994
640	1.13182 E-7	3.000	8.22735 E-7	2.998
1280	1.41486 E-8		1.03006 E-7	
160	1.44981 E-6	3.017	9.57814 E-6	2.981
320	1.79043 E-7	3.005	1.21318 E-6	2.997
640	2.23035 E-8	3.003	1.51957 E-7	2.999
1280	2.78216 E-9		1.90041 E-8	
160	7.09726 E-8	4.88	8.6567 E-7	4.97
320	2.41410 E-9	4.76	2.76804 E-8	3.98
640	8.92936 E-11	4.91	1.75192 E-9	3.48
1280	2.95859 E-12		1.36890 E-11	

TABLE 2.1: Burgers problem. Norm of the errors and convergence rate at $t = 0.3$ for the initial profile (2.44). The first group of values corresponds to the FDOC3 method with $\Delta t = 0.6\Delta x$. The second group corresponds to the FDOC5 method with the same time step. The third group corresponds again to the FDOC5 method, but with $\Delta t = 0.06\Delta x$.

equation contains a non-convex (s-shaped) flux of the form

$$f(u) = \frac{4u^2}{4u^2 + (1-u)^2}\,. \tag{2.45}$$

which means that the eigenvalues $\lambda(u)$ are not monotonic. The spectral radius in an interval is therefore not necessarily at one end of the interval. This is why our simple LLF implementation fails as we will see. Non-convex fluxes can lead to compound shock waves which are shocks adjacent to a rarefaction wave with wave speed equal to the shock speed at the point of attachment.

We will perform first a simulation with the initial data

$$u(x) = \begin{cases} 0 & 0 \le x < 1 - 1/\sqrt{2} \\ 1 & 1 - 1/\sqrt{2} \le x < 1 \end{cases} \tag{2.46}$$

so that the inflexion point in the flux (2.45) lies inside the interval spanned by the data.

The exact solution in this case is well approximated by a very-high-resolution (10.000 points) simulation using the first-order LLF algorithm, as displayed in Fig. 2.6 (continuous line). We see a right-propagating compound shock wave, consisting of a shock followed by a rarefaction wave, which propagates in the same direction. The results for our third-order algorithm, represented by the crosses line in Fig. 2.6, fail to reproduce correctly the rarefaction wave, which is replaced by an spurious intermediate state, resulting into a slower shock propagation speed.

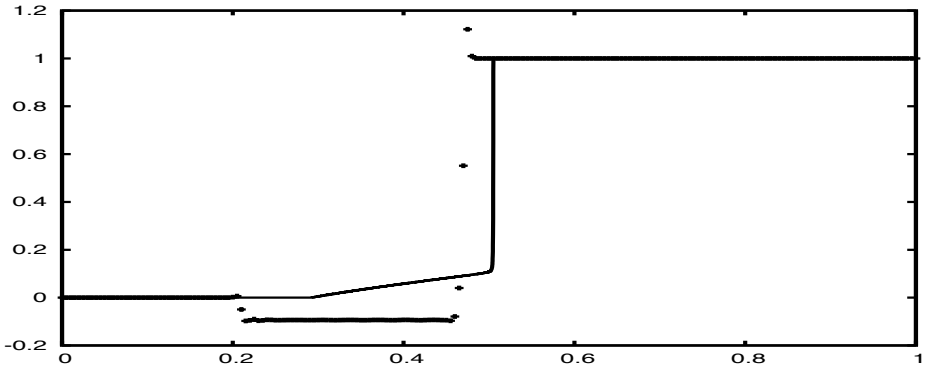


FIGURE 2.6: Buckley-Leverett's problem. The continuous line corresponds to the LLF first-order algorithm, with 10.000 points, as a replacement for the exact solution. The crosses line corresponds to the third-order algorithm FDOC3 with 200 points, converging towards a different solution.

In order to single out the problem, we have performed simulations for the same flux (2.45) but with a dynamical range that avoids the inflexion point either from below or from above. The results are plotted in Fig. 2.7, where we see either an ordinary rarefaction wave (left panel) or a simple shock (right panel), but no compound shock. In both cases, the third-order algorithm FDOC3 is able to model correctly the dynamics. This results indicate that the problem with compound shocks can be triggered by the presence of overshots at the connection point between the shock and the associated rarefaction wave, which can break the compound structure. The TVD character of the LLF flux prevents this problem to arise, as it is clearly shown in Fig. 2.6 (continuous line).

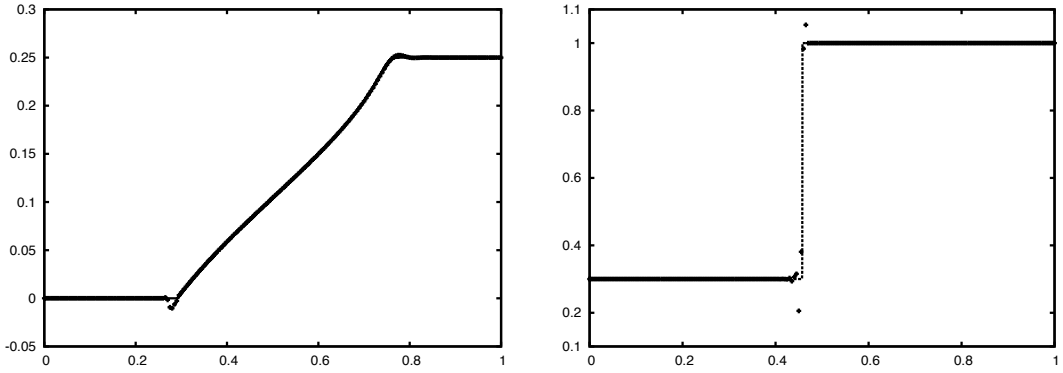


FIGURE 2.7: Same as in the previous figure, but now for two different dynamical ranges, which avoid the flux inflexion point. In the left panel, an ordinary rarefaction wave appears, which is correctly modelled by the third-order algorithm. In the right panel, a simple shock appears, well captured by the third-order algorithm.

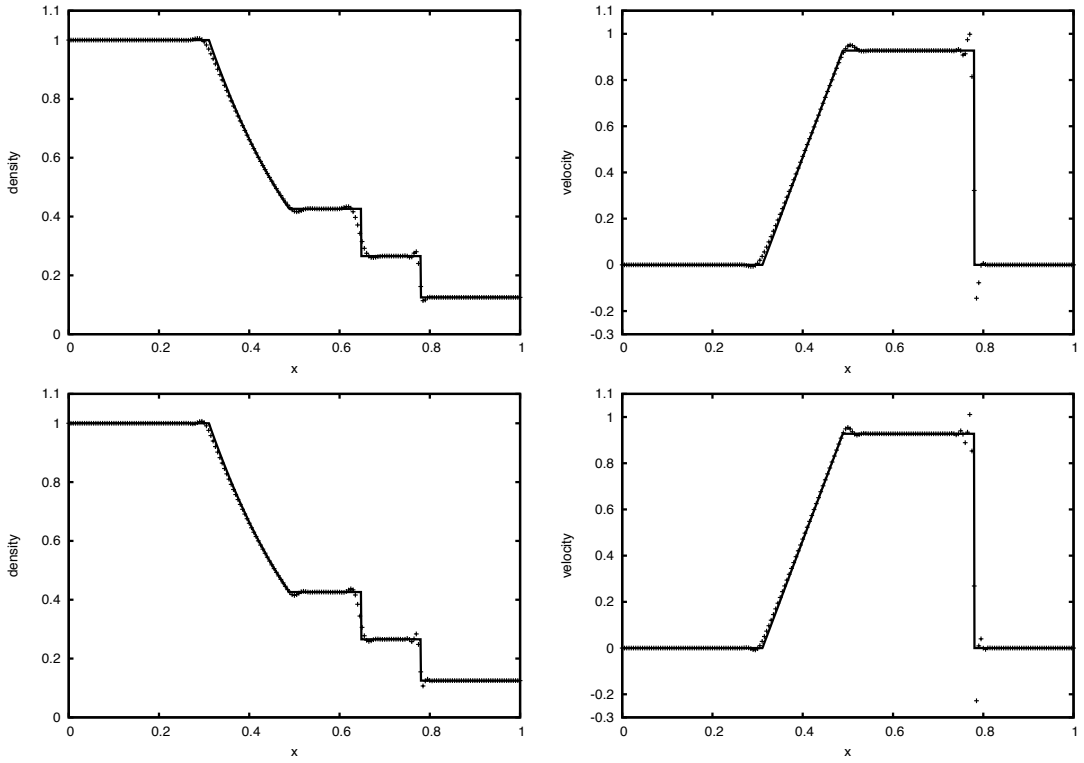


FIGURE 2.8: Sod shock tube problem. Density and speed profiles (left and right panels, respectively), for the ($m = 2$, $\beta = 1/12$) and the ($m = 3$, $\beta = 2/75$) schemes (upper and lower panels, respectively).

2.5.4 Euler equations

Euler equations for fluid dynamics are a convenient arena for testing the proposed schemes beyond the scalar case. In the ideal gas case, we can check the numerical results against well-known exact solutions containing shocks, contact discontinuities and rarefaction waves. We will deal first with the classical Sod shock-tube test [19] with a standard 200 points resolution.

We plot in Fig. 2.8 the gas density and speed profiles (left and right panels, respectively). Looking at the 3rd-order scheme results (upper panels), we see that both the rarefaction wave and the shock are perfectly resolved, whereas the contact discontinuity is smeared out. As a consequence, the main overshots are just besides the shock, specially visible in the speed profile, where the jump is much higher. Concerning the 5th-order scheme (lower panels), the contact discontinuity is slightly better resolved. This is however at the price of extra riddles and more visible overshots, so that the 3rd-order scheme seems to be more convenient.

A more demanding test is obtained when assuming a discontinuity in the initial speed, as in the Lax test [20]. As we see in Fig. 2.9, we get the same behavior than for the Sod

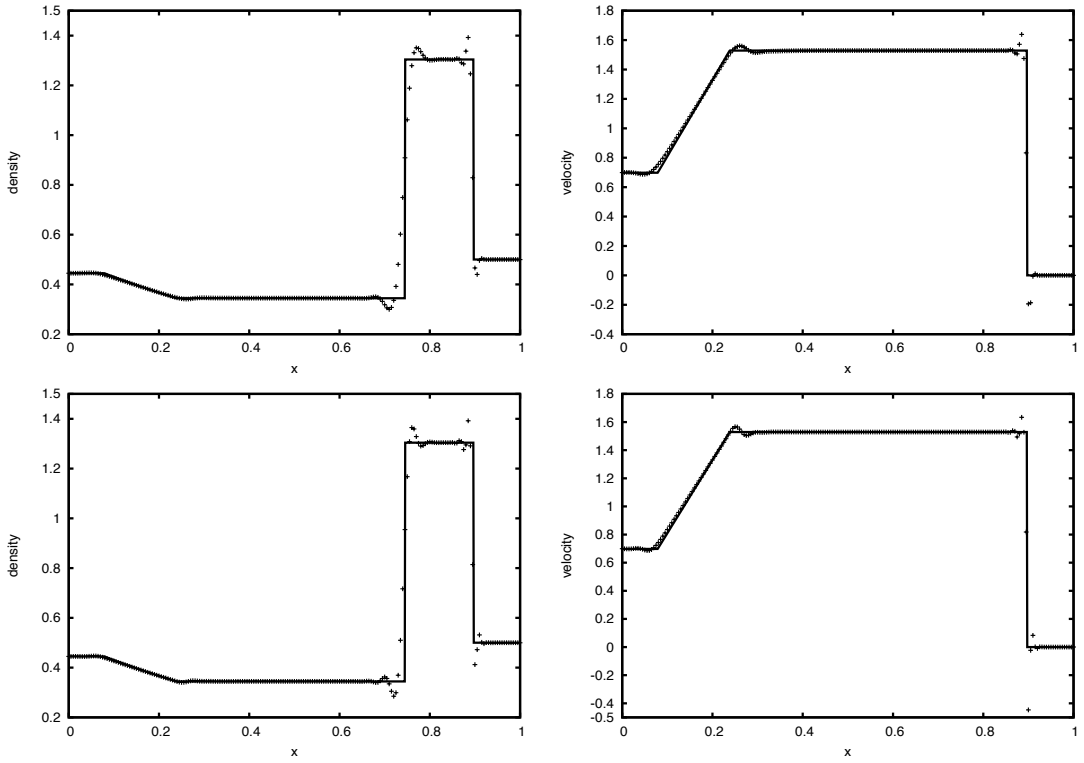


FIGURE 2.9: Lax shock tube problem. Density and speed profiles (left and right panels, respectively), for the FDOC3 and the FDOC5 schemes (upper and lower panels, respectively).

test case. The main difference is that the density jump at the contact discontinuity is much higher: the smearing of the density profile there is more visible, in contrast with the sharp shock profile nearby. Note also that some speed overshots are greater than the ones arising in the Sod test case (we have kept here the same 200 points resolution for comparison). The third-order algorithm seems to be more convenient again in this case.

2.6 Multidimensional tests

The results of this paper can be extended to a multidimensional case in a simple way. The semi-discrete equation (2.5) can be written in a rectangular grid as follows:

$$\partial_t u_{i,j} = -\frac{1}{\Delta x}\left(f_{i+1/2,j} - f_{i-1/2,j}\right) - \frac{1}{\Delta y}\left(f_{i,j+1/2} - f_{i,j-1/2}\right), \tag{2.47}$$

and the numerical flux can be computed by applying (2.12) to every single direction. Note however that the restriction (2.25) on the time step must be extended in this case

to

$$\lambda_j \, \Delta t \, (\frac{1}{\Delta x} + \frac{1}{\Delta y}) \, [\, d_{-1}^{\ m} - d_0^{\ m} + b \, \sum_{k \neq 0} max(d_{k-1}^m - d_k^{\ m}, 0) \,] \, \leq \, 1/2 \, . \tag{2.48}$$

In the finite-difference version (2.39), the extension to the multidimensional case amounts to replicate the right-hand-side difference operators for every single direction: no cross-derivative terms are required. This multidimensional extension allows to deal with some MHD tests, which add more complexity to the dynamics, clearly beyond the simple tests considered in the previous section.

2.6.1 The Orszag-Tang 2D vortex problem

As a first multi-dimensional example, let us consider here the Orszag-Tang vortex problem [21]. This is a well-known model problem for testing the transition to supersonic magnetohydrodynamical (MHD) turbulence and has become a common test of numerical MHD codes in two dimensions.

A barotropic fluid ($\gamma = 5/3$) is considered in a doubly periodic domain $[0, 2\pi]^2$, with uniform density ρ and pressure p. A velocity vortex given by $\mathbf{v} = (-\sin y, \sin x)$, corresponding to a Mach 1 rotation cell, is superimposed with a magnetic field $\mathbf{B} = (-\sin y, \sin 2x)$, describing magnetic islands with half the horizontal wavelength of the velocity roll. As a result, the magnetic field and the flow velocity differ in their modal structures along one spatial direction.

In Fig. 2.10 (upper panel) the temperature, $T = p/\rho$, is represented at a given time instant ($t = 3.14$). The figure clearly shows how the dynamics is an intricate interplay of shock formation and collision. The FDOC3 numerical scheme seems to handle the Orszag-Tang problem quite well. In Fig. 2.10 (lower panel) we plot the results for the same problem using a second order scheme built from the Roe-type solver and the monotonized-central (MC) symmetric limiter [22]. The results with both methods are qualitatively very similar.

2.6.2 Torrilhon MHD shock tube problem

We now consider the MHD shock tube problem described by Torrilhon [23] to investigate dynamical situations close to critical solutions. We will assume again a barotropic fluid with $\gamma = 5/3$. The initial conditions for the components of the magnetic field (B_2, B_3) are $(\cos\theta, \sin\theta)$, with $\theta = 0$ for $x \leq 0$. Depending on the angle θ between the left and right transverse components of the magnetic field, different types of solutions are found. Regular r-solutions consist only of shocks or contact discontinuities. Critical

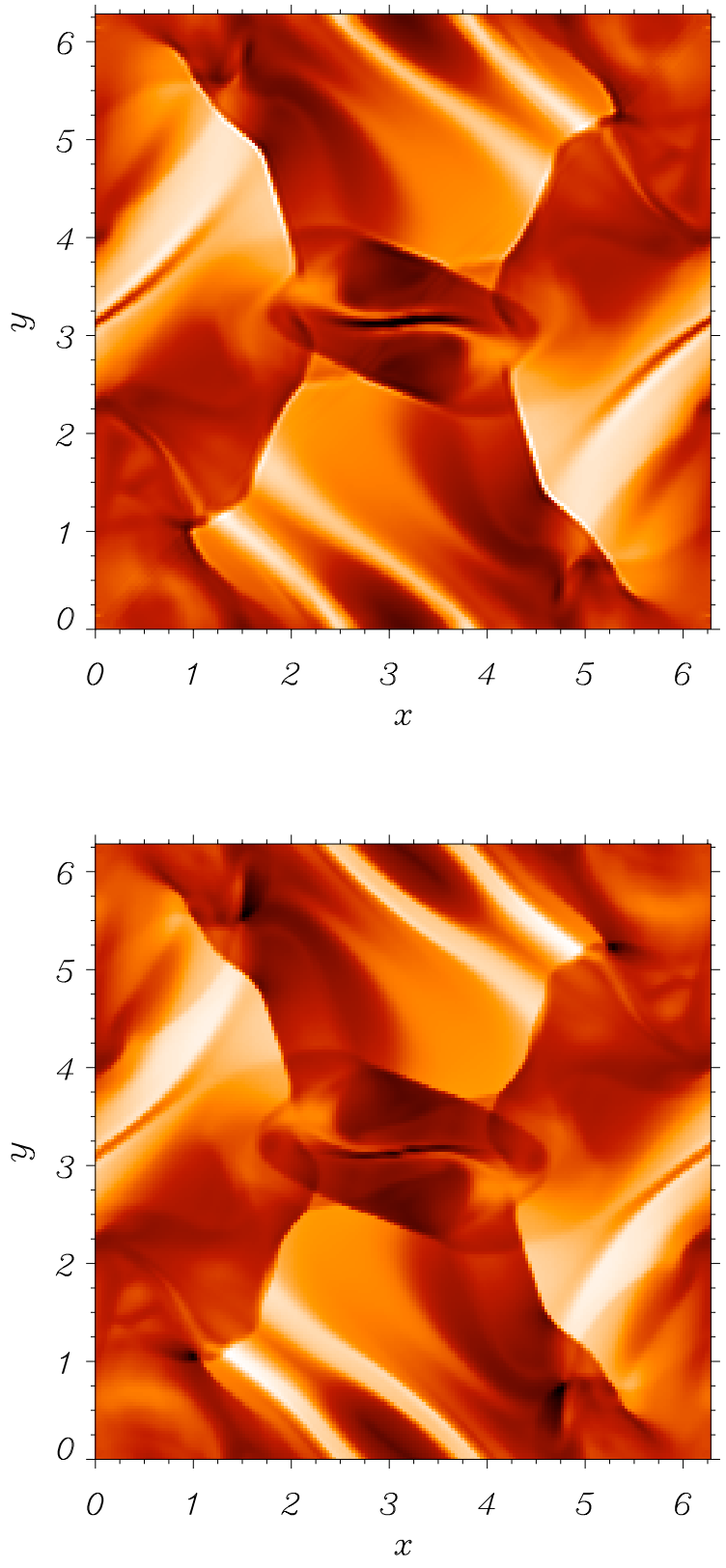


FIGURE 2.10: Temperature at $t = 3.14$ in the Orszag-Tang vortex test problem. In this simulation the grid has 200×200 mesh points. In the left panel the third-order scheme FDOC3 has been used while in the right panel the result is for a a second order scheme built from the Roe-type solver and the MC limiter.

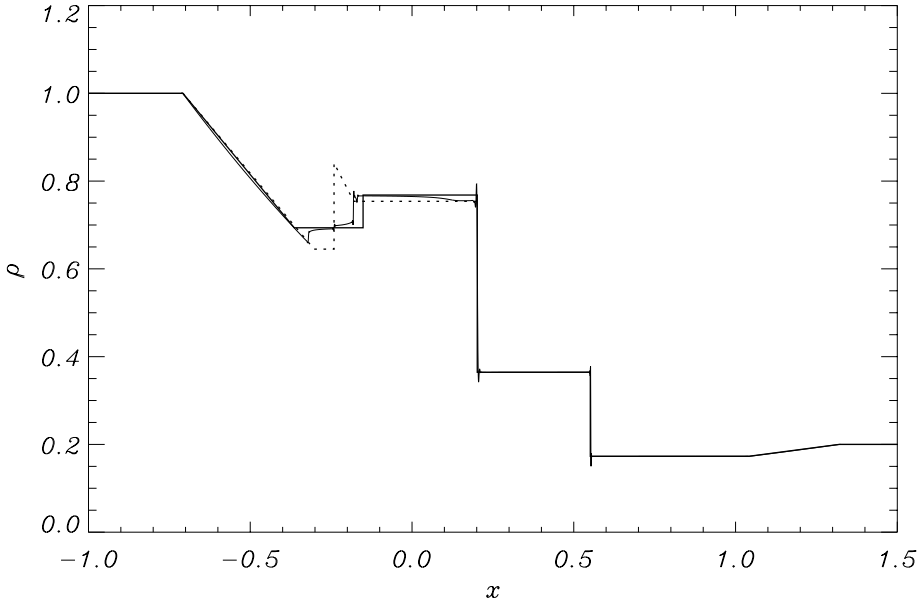


FIGURE 2.11: Plot of the density ρ at $t = 0.4$ for the almost co-planar problem with $\theta = 3$. In this simulation 5000 mesh points have been used. The dashed line represents the critical c-solution while the solid black line is the correct r-solution. Both solutions differ clearly in the interval $[-0.35, -0.1]$. The numerical simulation lies between the two.

c-solutions appear in the coplanar case, where the angle θ is an integer multiple of π. These solutions can contain also non-regular waves, such as compound waves.

We consider the situation for an *almost* co-planar case, $\theta = 3$. Analytically, this has a regular r-solution, but the numerical solution is attracted towards the nearby critical solution for $\theta = \pi$. Fig. 2.11 shows the density profile plotted together with the correct r-solution (solid black line) and the co-planar c-solution (dashed line). The r-solution has, from left to right, a rarefaction, a rotation, a shock, a contact discontinuity, a shock, a rotation and a rarefaction. The discrepancies among the different solutions are mainly in the interval $[-0.35, -0.1]$.

This interval is magnified in Fig. 2.12. The solid black line is the correct r-solution while the dashed line represents the critical c-solution. We see that the solutions with FDOC3 and FDOC5 tend to the correct solution although they keep some remnant from the c-solution. For comparison purposes we have also represented the numerical solution obtained with other schemes. We have used a second order LLF scheme and a second order Roe solver with either the minmod or the MC slope limiters.

The LLF scheme with the minmod limiter gets too close to the c-solution, even for this high-resolution simulation. The situation improves by replacing the minmod limiter by the MC one, but still gets farther from the right solution than the schemes proposed in

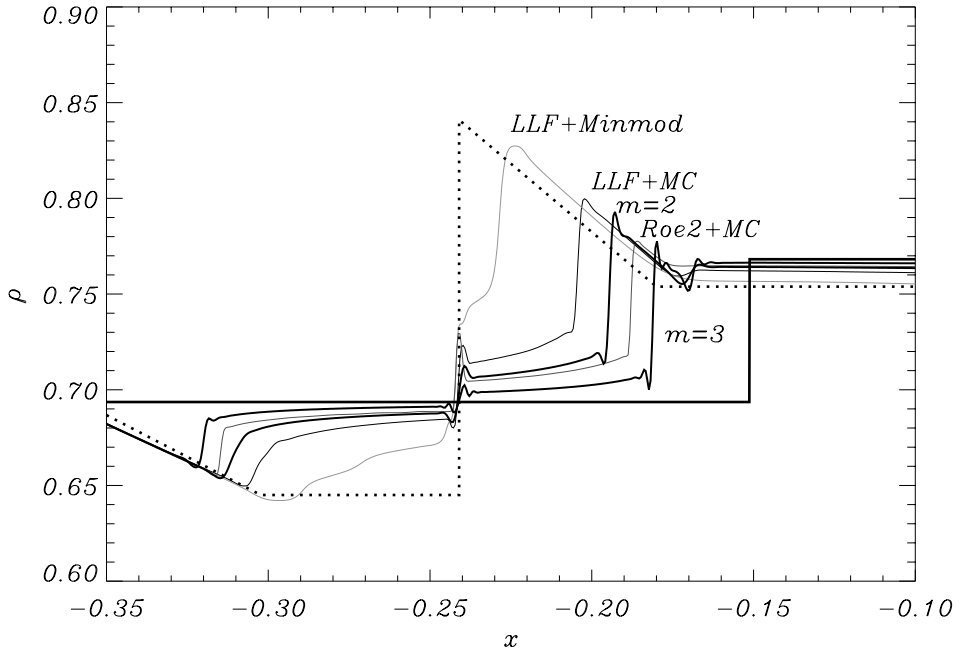


FIGURE 2.12: Same as Fig. 2.11, but enlarging the interval where the discrepancies show up. In addition to the exact regular and critical solutions, we plot, from top to bottom, the simulations for schemes using LLF with minmod limiter, LLF with MC limiter, the unlimited FDOC3 algorithm, a Roe solver with MC limiter and the unlimited FDOC5 algorithm.

this paper. Only the combination of a Roe-type solver with the MC limiter improves the results of the third-order scheme (FDOC3), but not those of the fifth-order scheme (FDOC5). This problem provides one specific example in which the fifth-order scheme seems to be more convenient than the third order one: the extra riddles are actually compensated by a clear improvement in the solution accuracy.

2.6.3 Double Mach reflection problem

This problem is a standard test case for high-resolution schemes. It corresponds to an experimental setting in which a shock is driven down a tube which contains a wedge. We will adopt here the standard configuration proposed by Woodward and Colella [24], involving a Mach 10 shock in air ($\gamma = 1.4$) at a 60^o angle with a reflecting wall. The air ahead of the shock is stationary with a density of 1.4 and a pressure of 1. The reflecting wall lies at the bottom of the computational domain, starting at $x = 1/6$. Allowing for this, the exact post-shock condition is imposed at the bottom boundary in the region $0 \leq x \leq 1/6$ and a reflecting wall condition is imposed for the rest. Inflow (post-shock) conditions are used at the left and top boundaries, whereas an outflow (no gradient) condition is used for the right boundary.

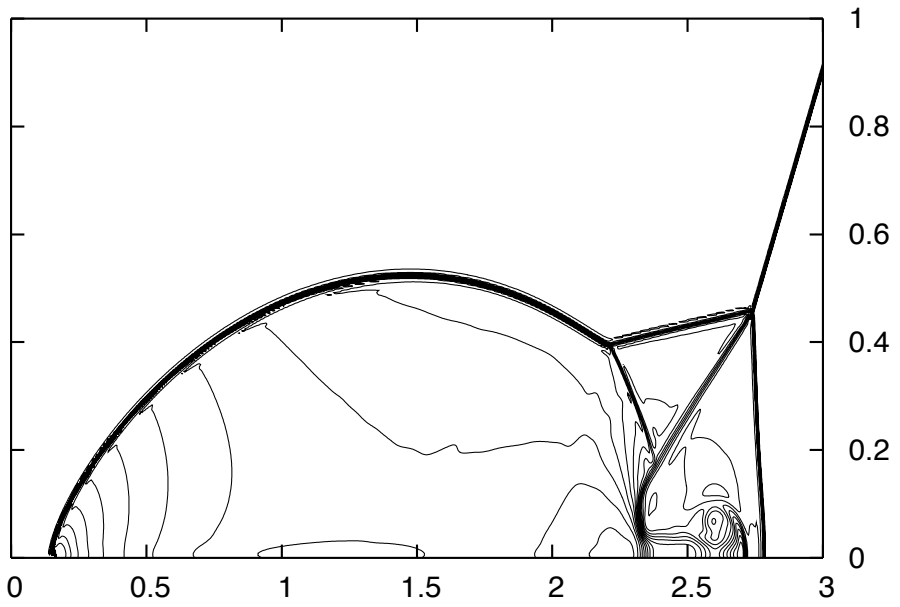
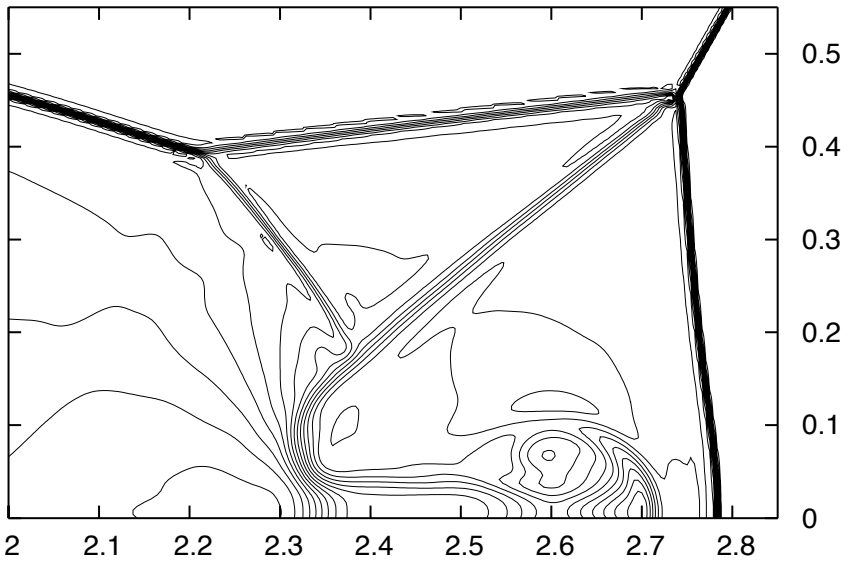
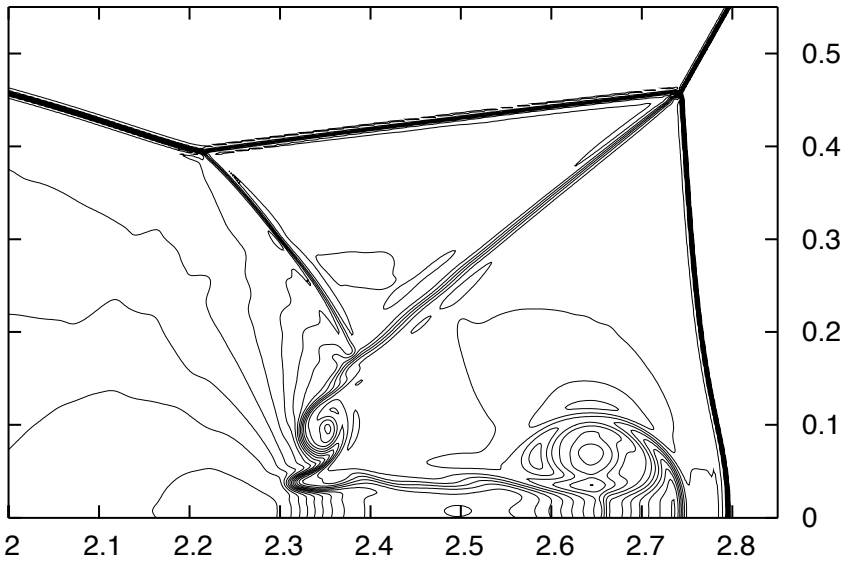


FIGURE 2.13: Double Mach reflection. Density plot at $t = 0.2$. The simulation is made with the 3rd-order method FDOC3 with $\Delta x = \Delta y = 1/240$. 30 evenly spaced density contours are shown.

This configuration leads to a complex flow structure, produced by a double Mach reflection of the shock at the wall. A self-similar flow (a fluid flow whose shape does not change with time) develops as the fluid meets the reflecting wall. Two Mach stems develop, with two contact discontinuities. We have plotted in Fig. 2.13 the density contours at $t = 0.2$, when the main features have fully developed. The more challenging ones are the jet propagating to the right near the reflecting wall and the weak shock generated at the second Mach reflection, as seen in the enlarged area in Fig. 2.14.

Our third-order results agree with the original ones [24] for the corresponding resolution: both the jet and the weak shock are clearly captured. Increasing both the resolution and the order-of-accuracy of the numerical algorithm, as shown in the subsequent panels in Fig. 2.14, we see more details of the jet rolling-up. Also, a vortex structure appears near the bottom wall, which starts affecting the diagonal contact discontinuity arising from the triple point. These high-resolution features, appearing in the last panel in Fig. 2.14, agree with the ones obtained with a WENO method of the same order (but double resolution, 1/960) in Ref. [25]. This also agrees with the results of recent spectral (finite) volume simulations [26], in which those structures show up gradually, as one is getting more accurate simulations. This is another example in which a higher-order algorithm can be preferred, as it captures more detailed features of complex structures for a given resolution.


0.5
0.4
0.3
0.2
0.1
0
2
2.1
2.2
2.3
2.4
2.5
2.6
2.7
2.8


0.5
0.4
0.3
0.2
0.1
0
2
2.1
2.2
2.3
2.4
2.5
2.6
2.7
2.8

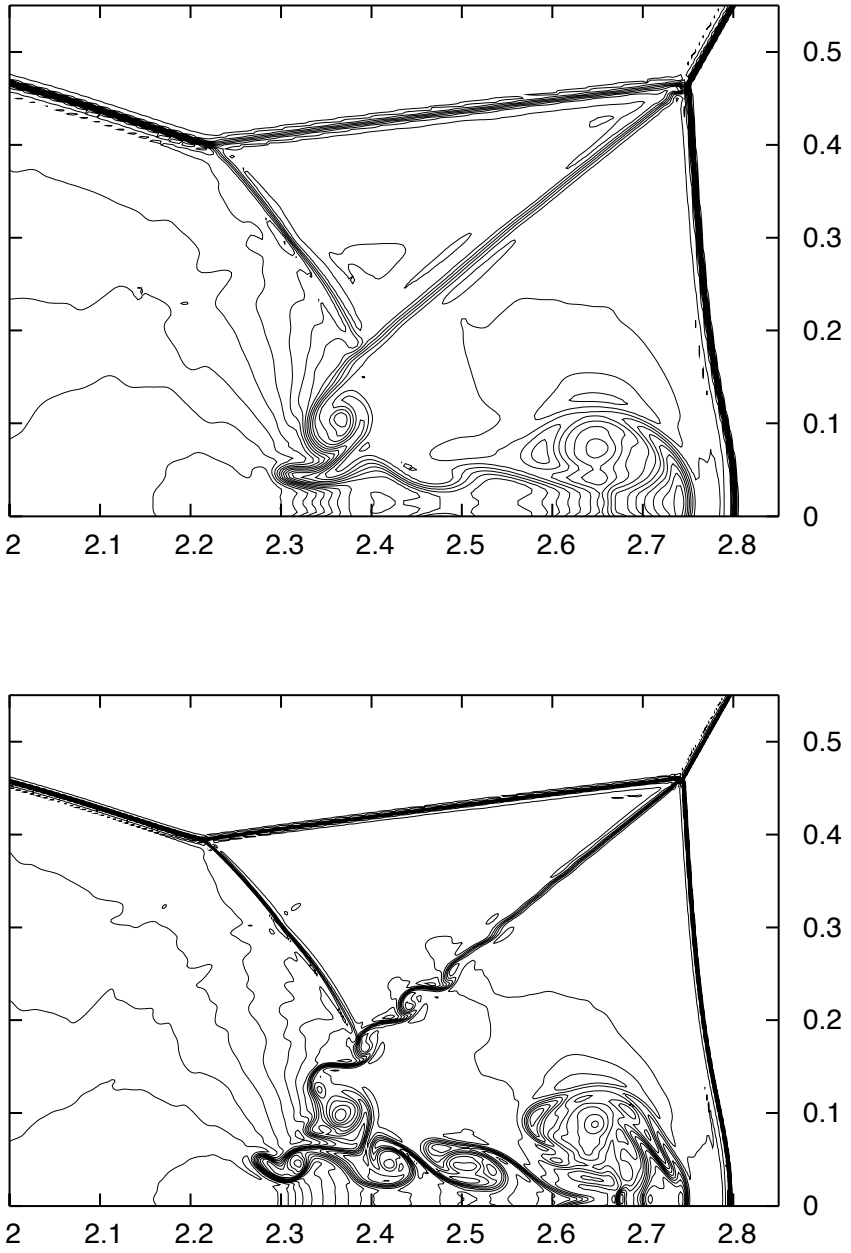


FIGURE 2.14: Same as Fig. 2.13, but enlarging the lower right corner. The panels on the previous page correspond to the third-order method FDOC3, with resolution of either 1/240 (top) or 1/480 (bottom). The panels in this page show the same for the 5th-order method FDOC5. Both the jet near the bottom wall and the weak shock, generated at the kink in the main reflected shock, are well resolved. A vortex structure at the bottom of the diagonal contact discontinuity shows up, with more details appearing when increasing accuracy.

2.7 Summary

The numerical experiments presented in this chapter provide clear evidence for a TVB behavior of the proposed schemes. This means that the total variation growth is uniformly bounded, independently of the resolution, for a fixed evolution time. Moreover, the experimental pattern is a sudden growth of the total variation, which provides a time-independent bound for the rest of the evolution. This growth is confined to the mesh points directly connected with non-sonic critical points, especially near discontinuities. But the resulting riddles do not spread over smooth regions and the overall features of the solution are preserved as a result. In the case of compound shocks, however, the numerical simulations actually mystify the physical solution: the spurious riddles affect the contact point between the shock and the adjacent rarefaction wave, breaking the compound structure, even if the TVB behavior is still preserved.

The proposed schemes are obtained from the unlimited version of the Osher-Chakrabarthy [8] linear flux-modification algorithms. The robustness of the unlimited version is related with the high compression factor of this algorithms family. We have actually improved the available estimates up to a remarkable value of $b = 5$, for the third-order case. This suggests that these estimates could be even improved by using alternative bound-setting procedures. Unfortunately, even in the scalar case, we are not able to prove rigorously the TVB properties of these methods.

We have combined the unlimited Osher-Chakrabarthy algorithm with the simple LLF flux formula. As a result, we have been able to derive the compact finite-difference scheme (2.39), which is equivalent to the corresponding finite-volume implementation in the unlimited case. This provides an extremely cost-efficient algorithm for dealing with the most common problems, even in presence of interacting dynamical shocks, as we have done in the Orszag-Tang 2D vortex and the double Mach reflection cases. Of course, its use should be limited to convex-flux problems, where compound shocks do not arise.

And, as we have already discussed, the resulting finite-difference formula (2.39) is similar to the ones obtained by the 'artificial viscosity' approach (see for instance ref. [27]). The main difference is that the spectral radius plays a key role here in the dissipation term, providing some sort of 'adaptive viscosity'. But the amount of viscosity is not arbitrary, as our compression factor estimates provide specific prescriptions for the value of the dissipation coefficient.

References

[1] L. Baiotti and L. Rezzolla, Phys. Rev. Lett. **97** 141101 (2006).

[2] B. Gustafson, H.O. Kreiss and J. Oliger, *Time dependent problems and difference methods*, Wiley, New York (1995).

[3] P. D. Lax, B. Wendroff (1960), "Systems of conservation laws". Commun. Pure Appl Math. 13: 217-237.

[4] R. W. MacCormack (1969), "The Effect of viscosity in hypervelocity impact cratering". AIAA Paper 69-354.

[5] S. K. Godunov (1959), "A Difference Scheme for Numerical Solution of Discontinuos Solution of Hydrodynamic Equations". Math. Sbornik 47: 271-306, translated US Joint Publ. Res. Service, JPRS 7226, (1969).

[6] V. V. Rusanov (1961), "Calculation of Intersection of Non-Steady Shock Waves with Obstacles". J. Comput. Math. Phys. USSR 1: 267-279.

[7] A. Harten (1983), "High Resolution Schemes for Hyperbolic Conservation Laws". J. Comput. Phys. 49: 357-393.

[8] S. Osher and S. Chakravarthy (1984), "Very High Order Accurate TVD schemes", ICASE Report 84-44, IMA Volumes in Mathematics and its Applications vol 2: 229-274. Springer-Verlag, 1986.

[9] Chi-Wang Shu (1987), "TVB Uniformly High-Order Schemes for Conservation Laws", Mathematics of Computation 49:105-121.

[10] A. Harten and S. Osher (1987), "Uniformly high-order accurate nonoscillatory schemes, I", SIAM J. Num. Anal. 24:279-309

[11] A. Harten, B. Engquist, S. Osher and S. Charavarty (1987), Uniformly high-order accurate essentially non-oscillatory schemes", J. Comp. Phys. 71:231-303.

[12] D. S. Balsara and Chi-Wang Shu (2000), "Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy", J. Comp. Phys. 160:405-452.

[13] Doron Levy, Gabriella Puppo and Giovanni Russo (2000), "A third order central WENO scheme for 2D conservation laws", Applied Numerical Mathematics 33:415.

[14] Steve Bryson and Doron Levy (2006), "On the total variation of High-Order Semi-Discrete Central schemes for Conservation Laws", Journal of Scientific Computation 27:163.

[15] A. Kurganov and E. Tadmor (2000), "New High-Resolution Central Schemes for Nonlinear Conservation Laws and Convection-Diffusion Equations", J. Comp. Phys. 160:214-282.

[16] A. Kurganov and Doron Levy (2000), "A Third-Order Semidiscrete Central Scheme for Conservation Laws and Convection-Diffusion Equations", SIAM J. Sci. Comput. 22:1461-1488.

[17] Xu-dong Liu and S. Osher (1996), "Nonoscillatory High Order Accurate Self-Similar Maximum Principle Satisfying Shock Capturing Schemes I", SIAM J. Numer. Anal. 33:439-471.

[18] R. J. LeVeque (1992), "Numerical Methods for Conservation Laws", Lectures in Mathematics, Birkhäuser.

[19] G. A. Sod (1978), "A Survey of Several Finite Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws". J. Comput. Phys. 27:1-31.

[20] P. D. Lax (1954), "Weak solutions of nonlinear hyperbolic equations and their numerical computation", Comm. Pure Appl. Math. 7:159193.

[21] S. A. Orszag and C. M. Tang (1979), "Small-scale structure of two-dimensional magnetohydrodynamic turbulence". J. Fluid Mech. 90, 1:129-143.

[22] B. Van Leer (1977), "Towards the ultimate conservative difference scheme III. Upstream-centered finite-difference schemes for ideal compressible flow". J. Comp. Phys. 23: 263-75.

[23] M. Torrilhon (2003), "Non-uniform convergence of finite volume schemes for Riemann problems of ideal magnetohydrodynamics". J. Comput. Phys. 192: 73-74.

[24] P. Woodward and P. Colella (1984), "The numerical simulation of two-dimensional fluid flow with strong shock". J. Comput. Phys. 54: 115173.

[25] J. Shi, Y-T. Zhan, and C-W. Shu (2003), "Resolution of high-order WENO schemes for complicated flow structures", J. Comput. Phys. 186: 690696.

[26] Z. J. Wang, L. Zhang and Y. Liu (2004), "Spectral finite volume method for conservation laws on unstructured grids IV: extension to two-dimensional systems", J. Comput. Phys. 194: 716741.

[27] B. Gustafsson, H. O. Kreiss and J. Oliger (1995), "Time Dependent Problems and Difference Methods". Wiley-Interscience (New York).

Chapter 3

Towards a gauge polyvalent numerical relativity code

3.1 Introduction

During these years, Kiuchi and Shinkai [1] have analyzed numerically the behavior of many 'adjusted' versions of the BSSN system. This is a follow-up of a former proposal [2] for using the energy-momentum constraints to modify Numerical Relativity evolution formalisms. An important point was to put the constraint propagation system (subsidiary system) in a strongly hyperbolic form, so that constraint violations can propagate out of the computational domain. As a further step, there is also the possibility of introducing damping terms, which would attract the numerical solution towards the constrained subspace.

At first sight, one could wonder why this idea is still deserving some interest today, when the BSSN system is being successfully used in binary-black-hole simulations. Waveform templates are currently being extracted for different mass and spin configurations, with an accuracy level that depends just on the computational resources (including the use of mesh-refinement and/or higher-order finite-difference algorithms). The same is true for neutron stars simulations, where the BSSN formalism is currently used for evolving the spacetime geometry [3]-[6]. But these success scenarios have a weak point: the BSSN simulations are based on the combination of the '1+log' and 'Gamma-driver' gauge conditions, as proposed in Ref. [7] for the first long-term dynamical simulation of a single Black Hole (BH) without excision.

Concerning BH simulations, we can understand that dealing numerically with collapse singularities requires the use of either excision, or time slicing prescriptions with strong

singularity-avoidance properties. In the '1+log' case, there is actually a 'limit hypersurface', so that the numerical evolution gets safely bounded away from collapse singularities. But singularity-avoidance is a property of the time coordinate, which should then be independent of the space coordinates prescription. In the spirit of General Relativity, we should expect a gauge-polyvalent numerical code to work as well in normal coordinates (zero shift), even if some specific type of time slicing condition (lapse choice) is required for BH simulations. Moreover, this requirement should be extended to other dynamical choices of the space coordinates. This means that a gauge-polyvalent numerical code should also work with alternative shift prescriptions, provided that the proposed choices preserve the regularity of the congruence of time lines. And this should be independent of the fact that a freezing of the dynamics is obtained or not as a result. These considerations apply *'a fortriori'* to neutron star simulations without any BH in the final stage, where no singularity is expected to form.

The above proposed gauge-polyvalence requirements, which are in keeping with the spirit of General Relativity, may seem too ambitious, allowing for the fact that they are not fulfilled by current BH codes (see Preface). But the need for improvement is even more manifest by looking at the results of the gauge-waves test. This test consists in evolving Minkowsky spacetime in non-trivial harmonic coordinates, and was devised for cross-comparing the numerical codes performance [8]. In Ref. [1], the authors assay different adjustments in order to correct the poor performance of 'plain' BSSN codes, which was previously reported in Ref. [9]. They manage to get long-term evolutions for the small amplitude case ($A = 0.01$) with a standard second-order-accurate numerical algorithm. The same result was previously achieved by using a fourth-order accurate finite differences scheme [10]. Even in this case, however, the results for the medium amplitude case ($A = 0.1$) are disappointing. More details can be found in a more recent cross-comparison paper [11], where actually a higher benchmark (big amplitude, $A = 0.5$, devised for testing the non-linear regime) is proposed.

One could argue that the gauge-waves test is not relevant for real simulations, because periodic boundary conditions do not allow constraint violations to propagate out of the computational domain [9]. In BH simulations, however, constraint violations arising inside the horizon can not get out, unless all the characteristic speeds of the subsidiary system are adjusted to be greater than light speed. As this extreme adjustment is not implemented in the current evolution formalisms, the gauge-waves test results can be indeed relevant, at least for non-excision BH codes. As a result, in keeping with the view expressed in Ref. [1], we are convinced that either an improvement of the current BSSN adjustments or any alternative formulation would be welcome, as it could contribute to widen the gauge-polyvalence of numerical relativity codes.

In this chapter we will consider an alternative numerical code consisting in two main ingredients. The first one is the Z4 strongly-hyperbolic formulation of the field equations [9]. The original (second order) version needs no adjustment for the energy and momentum constraints, as constraint deviations propagate with light speed, although some convenient damping terms have been also proposed [14]. We will present a first-order version, which has been adjusted for the ordering constraints which arise in the passage from the second-order to the first-order formalism. Its flux-conservative implementation is described in Appendix D. The second ingredient is the FDOC algorithm [14] presented in last chapter, which is a (unlimited) finite-difference version of the Osher-Chakrabarthy finite-volume algorithm [15], along the lines sketched in Chapter 2, which has been published in a previous paper [16]. Although this algorithm allows a much higher accuracy, we will restrict ourselves here to the simple cases of third and fifth-order accuracy, which have shown an outstanding robustness, confirmed by standard tests from Computational Fluid Dynamics, including multidimensional shock interactions [14].

The results for the gauge-waves test will be presented in this chapter, where just a small amount of dissipation, without any visible dispersion error, shows up after 1000 crossing times, even for the high amplitude ($A = 0.5$) case. Results from simulations of a 3D BH in normal coordinates will also be presented, where we will consider many variants of the 'Bona-Massó' singularity-avoidant prescription [17]. As expected, the best results for a given resolution are obtained for the choices with a limit hypersurface far away from the singularity. For the $f = 2/\alpha$ choice, the BH evolves in normal coordinates at least up to $1000\,M$ in a uniform grid with logarithmic space coordinates. This is one order of magnitude greater than the normal-coordinates BSSN result, as reported in [7].

Concerning the shift conditions, we have tested many explicit first-order prescriptions in single BH simulations. The idea is just to test the gauge-polyvalence of the code, so no physically motivated condition has been imposed, apart from the three-covariance of the shift under arbitrary time-independent coordinate transformations. Our results confirm that the proposed code is not specially tuned for normal coordinates (zero shift).

No sophisticated numerical tools (mesh refinement, algorithm-switching for the advection terms, etc) have been incorporated to our code at this point, when we are facing just test simulations. Concerning the boundary treatment, we simply choose at the points next to the boundary the most accurate centered algorithm compatible with the available stencil there. When it comes to the last point, we can either copy the neighbor value or propagate it out with the maximum propagation speed (by means of a 1D advection equation). The idea is to keep the numerical code as simple as possible in order to test here just the basic algorithm in a clean way.

3.2 Adjusting the first-order Z4 formalism

The Z4 formalism is a covariant extension of the Einstein field Equations, defined as [9]

$$R_{\mu\nu} + \nabla_\mu Z_\nu + \nabla_\nu Z_\mu = 8\,\pi\,(T_{\mu\nu} - \frac{1}{2}\,T\,g_{\mu\nu}). \tag{3.1}$$

The four vector Z_μ is an additional dynamical field, which evolution equations can be obtained from (3.1). The solutions of the original Einstein´s equations can be recovered when Z_μ is a Killing vector. In the generic case, the Killing equation has only the trivial solution $Z_\mu = 0$, so that true Einstein's solutions can be easily recognized.

The manifestly covariant form (3.1) can be translated into the 3+1 language in the standard way. The covariant four-vector Z_μ will be decomposed into its space components Z_i and the normal time component

$$\Theta \equiv n_\mu\,Z^\mu = \alpha\,Z^0 \tag{3.2}$$

where n_μ is the unit normal to the $t = constant$ slices. The 3+1 decomposition of (3.1) is given then by [9]:

$$(\partial_t - \mathcal{L}_\beta)\,\gamma_{ij} = -2\,\alpha\,K_{ij} \tag{3.3}$$

$$\begin{aligned}(\partial_t - \mathcal{L}_\beta)\,K_{ij} = &-\nabla_i\alpha_j + \alpha\,[R_{ij} + \nabla_i Z_j + \nabla_j Z_i \\ &- 2K_{ij}^2 + (tr\,K - 2\,\Theta)\,K_{ij} - 8\pi\{S_{ij} - \frac{1}{2}\,(tr\,S - \tau)\,\gamma_{ij}\}\,]\end{aligned} \tag{3.4}$$

$$(\partial_t - \mathcal{L}_\beta)\,\Theta = \frac{\alpha}{2}\,[R + 2\,\nabla_k Z^k + (tr\,K - 2\,\Theta)\,tr\,K - \mathrm{tr}\,(K^2) - 2\,Z^k\alpha_k/\alpha - 16\pi\tau] \tag{3.5}$$

$$(\partial_t - \mathcal{L}_\beta)\,Z_i = \alpha\,[\nabla_j\,(K_i{}^j - \delta_i{}^j\,tr\,K) + \partial_i\Theta - 2\,K_i{}^j\,Z_j - \Theta\,\alpha_i/\alpha - 8\pi S_i]\,. \tag{3.6}$$

The evolution system can be completed by providing suitable evolution equations for the lapse and shift components.

$$\partial_t\alpha = -\alpha^2\,Q\,, \qquad \partial_t\beta^i = -\,\alpha\,Q^i \tag{3.7}$$

We will keep open at this point the choice of gauge conditions, so that the gauge-derived quantities $\{Q,\,Q^i\}$ can be either a combination of the other dynamical fields or independent quantities with their own evolution equation. We are assuming, however, that both lapse and shift are dynamical quantities, so that terms involving derivatives of $\{Q,\,Q^i\}$ actually belong to the principal part of the evolution system.

First-order formulation: ordering constraints

In order to translate the evolution system (3.3-3.7) into a fully first-order form, the space derivatives of the metric components (including lapse and shift) must be introduced as new independent quantities:

$$A_i \equiv \partial_i \ln\alpha, \quad B_k{}^i \equiv \partial_k \beta^i, \quad D_{kij} \equiv \frac{1}{2}\,\partial_k \gamma_{ij}\ . \tag{3.8}$$

Note that, as far as the new quantities will be computed now through their own evolution equations, the original definitions (3.8) must be considered rather as constraints (first-order constraints), namely

$$\mathcal{A}_k \equiv A_k - \partial_k\, ln\alpha = 0 \tag{3.9}$$

$$\mathcal{B}_k{}^i \equiv B_k{}^i - \partial_k\, \beta^i = 0 \tag{3.10}$$

$$\mathcal{D}_{kij} \equiv D_{kij} - \frac{1}{2}\,\partial_k \gamma_{ij} = 0\,. \tag{3.11}$$

Note also that we can derive in this way the following set of constraints, related with the ordering of second derivatives (ordering constraints):

$$\mathcal{C}_{ij} \equiv \partial_i\, \mathcal{A}_j - \partial_j\, \mathcal{A}_i = \partial_i\, A_j - \partial_j\, A_i = 0\,, \tag{3.12}$$

$$\mathcal{C}_{rs}{}^i \equiv \partial_r\, \mathcal{B}_s{}^i - \partial_s\, \mathcal{B}_r{}^i = \partial_r\, B_s{}^i - \partial_s\, B_r{}^i = 0\,, \tag{3.13}$$

$$\mathcal{C}_{rsij} \equiv \partial_r\, \mathcal{D}_{sij} - \partial_s\, \mathcal{D}_{rij} = \partial_r\, D_{sij} - \partial_s\, D_{rij} = 0\,. \tag{3.14}$$

The evolution of the lapse and shift space derivatives could be obtained easily, just by taking the time derivative of the definitions (3.8) and exchanging the order of time and space derivatives. But then the characteristic lines for the transverse-derivative components in (3.8) would be the time lines (zero characteristic speed). This can lead to a characteristic degeneracy problem, because the characteristic cones of the second-order system (3.4-3.6) are basically the light cones [9], and the time lines can actually cross the light cones, as it is the case in many black hole simulations. In order to avoid this degeneracy problem, we can make use of the shift ordering constraint (3.13) for obtaining the following evolution equations for the additional quantities (3.8):

$$\partial_t A_k + \partial_l[-\beta^l\ A_k + \delta^l{}_k\ (\alpha\ Q + \beta^r A_r)] = B_k{}^l\ A_l - trB\ A_k \tag{3.15}$$

$$\partial_t B_k{}^i + \partial_l[-\beta^l\ B_k{}^i + \delta^l{}_k\ (\alpha\ Q^i + \beta^r B_r{}^i)] = B_k{}^l\ B_l{}^i - trB\ B_k{}^i \tag{3.16}$$

$$\partial_t D_{kij} + \partial_l[-\beta^l D_{kij} + \delta^l{}_k\ \{\alpha\ K_{ij} - 1/2\ (B_{ij} + B_{j\,i})\}\,] = B_k{}^l\ D_{lij} - trB\ D_{kij}\ . \tag{3.17}$$

Note that the characteristic lines for the transverse-derivative components are now the normal lines (instead of the time lines), so that characteristic crossing is actually avoided. This ordering adjustment is crucial for long-term evolution in the dynamical shift case, as it has been yet realized in the first-order version of the generalized harmonic formulation [18].

Damping terms adjustments

A further adjustment could be the introduction of some constraint-violation damping terms. For the energy-momentum constraints, these terms can be added to the evolution equations (3.4-3.6), as described in Ref. [14].

For the ordering constraints, we can also introduce simple constraint-violation damping terms when required. For instance, equation (3.15) could be modified as follows:

$$\partial_t A_i + \partial_l[-\beta^l A_i + \delta^l{}_i\ (\alpha\ Q + \beta^r A_r)] = B_i{}^l\ A_l - trB\ A_i - \eta\ \mathcal{A}_i\ , \tag{3.18}$$

with the damping parameter in the range $0 \leq \eta \ll 1/\Delta t$. The same pattern could be applied to equations (3.16, 3.17).

In order to justify this, let us analyze the resulting evolution equations for the first-order constraints (3.9). Allowing for (3.15), we would get

$$\partial_t\, \mathcal{A}_k - \beta^r\, (\partial_r\, \mathcal{A}_k - \partial_k\, \mathcal{A}_r) = \mathcal{B}_k{}^r\, A_r - \mathcal{B}_r{}^r\, A_k\, . \tag{3.19}$$

The hyperbolicity of the subsidiary evolution equation (3.19) can be analyzed by looking at the normal and transverse components of the principal part along any space direction $\vec{n}$, namely

$$\partial_t\, \mathcal{A}_n - \beta^\perp\, (\partial_n\, \mathcal{A}_\perp) = 0 \tag{3.20}$$

$$\partial_t\, \mathcal{A}_\perp - \beta^n\, (\partial_n\, \mathcal{A}_\perp) = 0\, , \tag{3.21}$$

with eigenvalues $(0,\ -\beta^n)$, which is just weakly hyperbolic in the fully degenerate case, that is for any space direction orthogonal to the shift vector. Note that this is just the subsidiary system governing constraint violations, not the evolution system itself. This means that the main concern here is accuracy, rather than stability. But the resulting (linear) secular growth of first-order constraint violations may become unacceptable in long-term simulations.

These considerations explain the importance of adding constraint-damping terms, so that (3.15) is replaced by (3.18). The damping term $-\eta\, \mathcal{A}_k$ will appear as a result in the

subsidiary system also. The linearly growing constraint-violation modes arising from the degenerate coupling in (3.20) will be kept then under control by these (exponential) damping terms. The same argument applies *mutatis mutandis* to the remaining first-order constraints $\mathcal{B}_k{}^i$, $\mathcal{D}_{kij}$.

Secondary ordering ambiguities

The shift ordering constraints (3.13) can also be used for modifying the first-order version of the evolution equation (3.6) in the following way

$$(\partial_t - \mathcal{L}_\beta)\, Z_i = \alpha\,[\,\nabla_j\,(K_i{}^j - \delta_i{}^j\, tr\, K) + \partial_i\,\Theta - 2\,K_i{}^j\, Z_j - \Theta\, A_i - 8\pi S_i\,] - \mu\,(\partial_j\, B_i{}^j - \partial_i\, tr B)\,. \tag{3.22}$$

Also, the ordering constraints (3.14) can be used for selecting a specific first-order form for the three-dimensional Ricci tensor appearing in (3.4) [19]. This can be any combination of the standard Ricci decomposition

$$R_{ij} \;=\; \partial_k\,\Gamma^k{}_{ij} - \partial_i\,\Gamma^k{}_{kj} + \Gamma^r{}_{rk}\,\Gamma^k{}_{ij} - \Gamma^k{}_{ri}\,\Gamma^r{}_{kj} \tag{3.23}$$

with the De Donder decomposition

$$\begin{aligned} R_{ij} \;&=\; -\partial_k\, D^k{}_{ij} + \partial_{(i}\,\Gamma_{j)k}{}^k - 2 D_r{}^{rk} D_{kij} \\ &+\; 4\, D^{rs}{}_i D_{rsj} - \Gamma_{irs}\Gamma_j{}^{rs} - \Gamma_{rij}\,\Gamma^{rk}{}_k \end{aligned} \tag{3.24}$$

which is most commonly used in Numerical Relativity codes. Following Ref. [19], we will introduce an ordering parameter ξ, so that $\xi = 1$ corresponds to the Ricci decomposition (C.6) and $\xi = -1$ to the De Donder one (3.24).

The choices of μ and ξ do not affect the characteristic speeds of the evolution system (see Appendix D for details), nor the structure of the subsidiary system. In this sense, these are rather secondary ordering ambiguities and we will keep these parameters free for the moment, although there are some prescriptions that can be theoretically motivated:

- The choice $\mu = 1/2,\ \xi = -1$ allows to recover at the first-order level the equivalence between the generalized harmonic formulation and (the second-order version of) the Z4 formalism, given by [14]

$$Z^\mu = \frac{1}{2}\,\Gamma^\mu{}_{\rho\sigma}\, g^{\rho\sigma} \tag{3.25}$$

  (see Appendix D for more details). This can be important, because the harmonic system is known to be symmetric hyperbolic.

- The choice $\mu = 1$ is the only one that ensures the strong hyperbolicity of the Z3 system, obtained from the Z4 one by setting $\theta = 0$. This can be relevant if we are trying to keep energy-constraint violations close to zero. Allowing for the quasi-equivalence between the Z3 and the BSSN systems [19], this adjustment will affect as well to the first-order version of the BSSN system (NOR system [20]) in simulations using dynamical shift conditions. The same comment applies to the old 'Bona-Massó' system [21].

- The choice $\xi = 0$ ensures that the first-order version contains only symmetric combinations of second derivatives of the space metric. This is a standard symmetrization procedure for obtaining a first-order version of a generic second-order equation.

In the numerical simulations in this chapter, we have taken $\mu = 1,\ \xi = -1$, although we have also tested other combinations, which also lead to long-term stability.

3.3 Gauge waves test

We will begin with a test devised for harmonic coordinates. Let us consider the following line element:

$$ds^2 = H(x-t)(-dt^2 + dx^2) + dy^2 + dz^2 \ , \tag{3.26}$$

where H is an arbitrary function of its argument. One could naively interpret this as the propagation of an arbitrary wave profile with unit speed. But it is a pure gauge effect, because (3.26) is nothing but the Minkowsky metric, written in some non-trivial harmonic coordinates system.

As proposed in Refs. [8], [11], we will consider the 'gauge waves' line element (3.26), with the following profile:

$$H = 1 - A\ Sin(\ 2\pi(x-t)\ )\,, \tag{3.27}$$

so that the resulting metric is periodic and we can identify for instance the points -0.5 and 0.5 on the x axis. This allows to set up periodic boundary conditions in numerical simulations, so that the initial profile keeps turning around along the x direction. One can in this way test the long term effect of these gauge perturbations. The results show that the linear regime (small amplitude, $A = 0.01$) poses no serious challenge to most Numerical Relativity codes (but see Ref. [1] for the BSSN case). Following the recent suggestion in Ref. [11], we will then focus in the medium and big amplitude cases ($A = 0.1$ and $A = 0.5$, respectively), in order to test the non-linear regime.

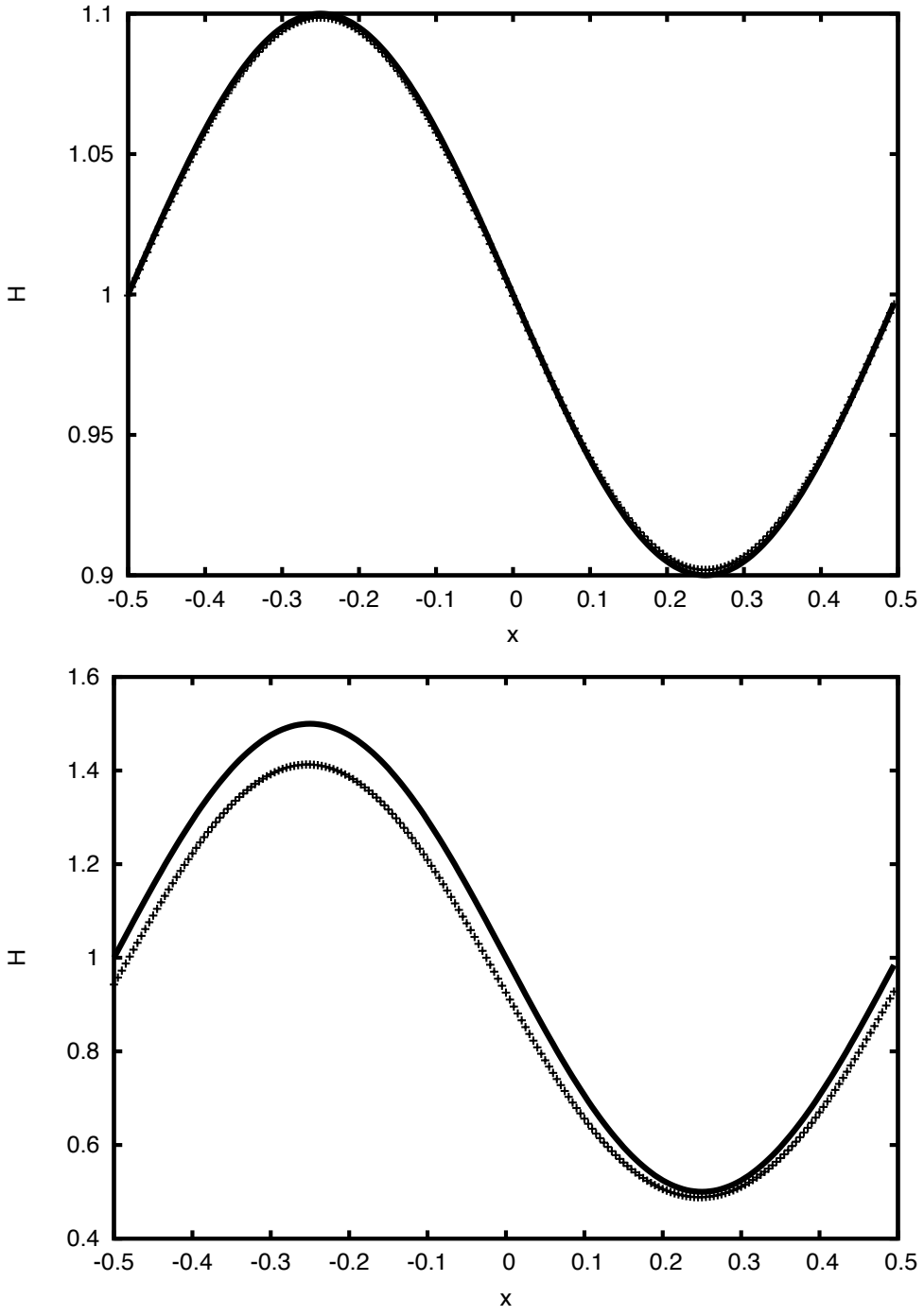


FIGURE 3.1: Gauge waves simulation with periodic boundary conditions and sinusoidal initial data for the γ_{xx} metric component. The resolution is $\Delta x = 0.005$ in both cases. The upper panel corresponds to the medium amplitude case $A = 0.1$. After 1000 round trips, the evolved profile (cross marks) nearly overlaps the initial one (continuous line), which corresponds also with the exact solution. The lower panel corresponds to the same simulation for the big amplitude case $A = 0.5$. We see the combination of a slight decrease in the mean value plus some amplitude damping.

Concerning grid spacing, although $\Delta x = 0.01$ would be enough for passing the test in the medium amplitude case, the big amplitude one requires more resolution, so we have taken $\Delta x = 0.005$ in both cases.

The results of the numerical simulations are displayed in Fig. 3.1 for the H function (the γ_{xx} metric component). The left panel shows the medium amplitude case $A = 0.1$. Only a small amount of numerical dissipation is barely visible after 1000 round trips: the third-order-accurate finite-difference method gets rid of the dominant dispersion error. For comparison, let us recall that the corresponding BSSN simulation crashes before 100 round trips [10]. The right panel shows the same thing for the large amplitude case $A = 0.5$, well inside the non-linear regime. We see some amplitude damping, together with a slight decrease of the mean value of the lapse.

Our results are at the same quality level than the ones reported in Ref. [11] for the Flux-Conservative generalized-harmonic code Abigail (see also the 'apples with apples' webpage [22]), which is remarkable for a test running in strictly harmonic coordinates. We can also compare with the simulations reported in Ref. [23] for (a specific variant of) the KST evolution system [24]. Although the gauge wave parametrization is not the standard one, both their 'big amplitude' case and their finest resolution are similar to ours. We see a clear phase shift, due to cumulative dispersion errors, after about 500 crossing times. We see also a growing amplitude mode, which can be moderated with resolution (for the finest one, it just compensates numerical dissipation). This can be related with the spurious linear mode that has been reported for harmonic systems which are not written in Flux-Conservative form [8].

We can conclude that there are two specific ingredients in our code that contribute to the gauge-wave results in an essential way: the Flux-Conservative form of the equations (see Appendix D), which gets rid of the spurious growing amplitude modes, and the third-order accuracy of the numerical algorithm, which reduces the dispersion error below the visual detection level in Fig. 3.1, even after 1000 crossing times.

3.4 Single Black hole test: normal coordinates

We will try next to test a Schwarzschild black-hole evolution in normal coordinates (zero shift). Harmonic codes are not devised for this gauge choice, so we will compare with BSSN results instead. Concerning the time coordinate condition, our choice will be limited by the singularity-avoidance requirement, as far as we are not going to excise the black-hole interior. Allowing for these considerations, we will determine the gauge

evolution equations (3.7) as follows

$$Q = f\ (tr\,K - m\,\Theta)\ , \qquad Q^i = 0\ \ (\beta^i = 0)\ , \tag{3.28}$$

where the second gauge parameter m is a feature of the Z4 formalism. We will choose here by default $m = 2$, because the evolution equation for the combination $trK - 2\,\Theta$, as derived from (3.4, 3.5), actually corresponds with the BSSN evolution equation for $tr\,K$ (see Ref. [19] for the relationship between BSSN and Z4 formalisms).

Concerning the first gauge parameter, we will consider first the '1+log' choice $f = 2/\alpha$ [25], which is the one used in current binary BH simulations in the BSSN formalism. The name comes from the resulting form of the lapse, after integrating the evolution equation (3.3, 3.7) with the prescription (3.28) for true Einstein's solutions ($\Theta = 0$):

$$\alpha = \alpha_0 + ln\,(\gamma/\gamma_0)\ , \tag{3.29}$$

where $\sqrt{\gamma}$ is the space volume element. It follows from (3.29) that the coordinate time evolution stops at some limit hypersurface, before even getting close to the collapse singularity. This happens when

$$\sqrt{\gamma/\gamma_0} = exp\,(-\alpha_0/2)\ , \tag{3.30}$$

that is well before the vanishing of the space volume element: the initial lapse value is usually close to one, so that the final volume element is still about a 60% of the initial one. This can explain the robustness of the 1+log choice in current black-hole simulations.

We will consider as usual initial data on a time-symmetric time slice ($K_{ij} = 0$) with the intrinsic metric given in isotropic coordinates:

$$\gamma_{ij} = (1 + \frac{m}{2r})^4\ \delta_{ij}\ . \tag{3.31}$$

This is the usual 'puncture' metric, with the apparent horizon at $r = m/2$: the interior region is isometric to the exterior one, so that the $r = 0$ singularity is actually the image of space infinity. We prefer, however, to deal with non-singular initial data. We will then replace the constant mass profile in interior region $r < M/2$ by a suitable profile $m(r)$, so that the interior metric corresponds to a scalar field matter content. Of course, the scalar field itself must be evolved consistently there (see Appendix E for details). A previous implementation of the same idea, with dust interior metrics, can be found in Ref. [26].

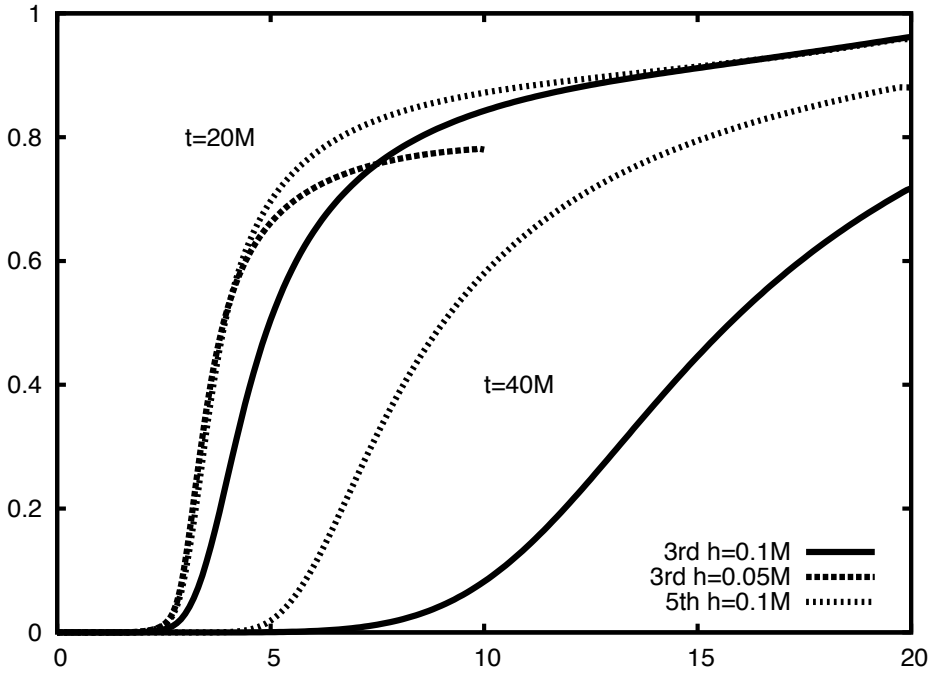


FIGURE 3.2: Plots of the lapse profiles at $t = 20M$ and $t = 40\,M$. The results for the third-order accurate algorithm (continuous lines) are compared with those for the fifth-order algorithm (dotted lines) for the same resolution ($h = 0.1M$). We have also included for comparison one extra line, corresponding to the third-order results with $h = 0.05\,M$, computed in a reduced mesh. Increasing resolution leads to a slope steepening and a slower propagation of the collapse front. In this sense, as we can see for $t = 20\,M$, switching to the fifth-order algorithm while keeping $h = 0.1\,M$ amounts to doubling the resolution for the third-order algorithm.

We have performed a numerical simulation for the $f = 2/\alpha$ case with a uniform grid with resolution $h = 0.1\,M$, extending up to $r = 20\,M$ (no mesh-refinement). We have used the third and fifth-order FDOC algorithms, as described in previous chapter, with the optimal dissipation parameters for each case. The results for the lapse profile are shown in Fig. 3.2 at $t = 20\,M$ an $t = 40\,M$. We see in both cases that the higher order algorithm leads to steeper profiles and a slower propagation of the collapse front. Note that the differences in the front propagation speed keep growing in time, although the third-order plot at $t = 40\,M$ is clearly affected by the vicinity of the outer boundary. This fact does not affect the code stability, as far as we can proceed with the simulations beyond $t = 50\,M$, when the collapse front gets out of the computational domain (beyond $t = 60\,M$ in the higher-order simulations). Note that the corresponding BSSN simulations ($f = 2/\alpha$ in normal coordinates) are reported to crash at about $t = 40\,M$ [7].

We have added for comparison an extra plot in Fig. 3.2, with the results at $t = 20\,M$ of a third-order simulation with double resolution ($h = 0.05\,M$), obtained in a smaller computational domain (extending up to $10\,M$). Both the position and the slope of the collapse front coincide with those of the fitfth-order algorithm with $h = 0.1\,M$. In

this case, switching to the higher-order algorithm amounts to doubling accuracy. Note, however, that higher-order algorithms are known to be less robust [14]. Moreover, as the profiles steepen, the risk of under-resolution at the collapse front increases. We have found that a fifth-order algorithm is a convenient trade-off for our $h = 0.1\,M$ resolution in isotropic coordinates.

We have also explored other slicing prescriptions with limit surfaces closer to the singularity, as described in Table 3.1. Note that in these cases the collapse front gets steeper than the one shown in Fig. 3.2 for the standard $f = 2/\alpha$ case with the same resolution. This poses an extra challenge to numerical algorithms, so we have switched to the third-order-accurate one for the sake of robustness. In all cases, the simulations reached $t = 50\,M$ without problem, meaning that the collapse front has get out of the computational domain. It follows that the standard prescription $f = 2/\alpha$, although it leads actually to smoother profiles, is not crucial for code stability.

f	$2/\alpha$	$1+1/\alpha$	$1/2+1/\alpha$	$1/\alpha$
$\sqrt{\gamma/\gamma_0}$	61%	50%	44%	37%

TABLE 3.1: Different prescriptions for the gauge parameter f, with the corresponding values of the residual volume element at the limit surface (normal coordinates), assuming a unit value of the initial lapse.

The results shown in Fig. 3.2 compare with the ones in Ref. [27], obtained with (a second-order version of) the old Bona-Massó formalism. We see the same kind of steep profiles, produced by the well known slice-stretching mechanism [28]. This poses a challenge to standard numerical methods: in Ref. [27] Finite-Volume methods where used, including slope limiters. Our FDOC algorithm (see Ref. [14] or Chapter 2 for details) can also be interpreted as an efficient Finite-Differences (unlimited) version of the Osher-Chakrabarthy Finite-Volume algorithm [15]. Note however that in Ref. [27], like in the BSSN case, a conformal decomposition of the space metric was considered, and an spurious (numerical) trace mode arise in the trace-free part of the extrinsic curvature. An additional mechanism for resetting this trace to zero was actually required for stability. In our (first-order) Z4 simulations, both the plain space metric and extrinsic curvature can be used directly instead, without requiring any such trace-cleaning mechanisms.

Let us take one further step. Note that the lifetime of our isotropic coordinates simulations (with no shift) is clearly limited by the vicinity of the boundary (at $r = 20\,M$). At this point, we can appeal to space coordinates freedom, switching to some logarithmic coordinates, as defined by

$$r = L\ sinh(R/L)\ , \tag{3.32}$$

where R is the new radial coordinate and L some length scale factor. This configuration suggests using the third-order algorithm FDOC3 because of its higher robustness. We

have performed a long-term numerical simulation for the $f = 2/\alpha$ case, with $L = 1.5\,M$, so that $R = 20\,M$ in these logarithmic coordinates corresponds to about $r = 463.000\,M$ in the original isotropic coordinates. In this way, as shown in Fig. 3.3, the collapse front is safely away from the boundary, even at very late times. We stopped our code at $t = 1000\,M$, without any sign of instability. This provides a new benchmark for Numerical Relativity codes: a long-term simulation of a single black-hole, without excision, in normal coordinates (zero shift). Moreover, it shows that a non-trivial shift prescription is not a requisite for code stability in BH simulations.

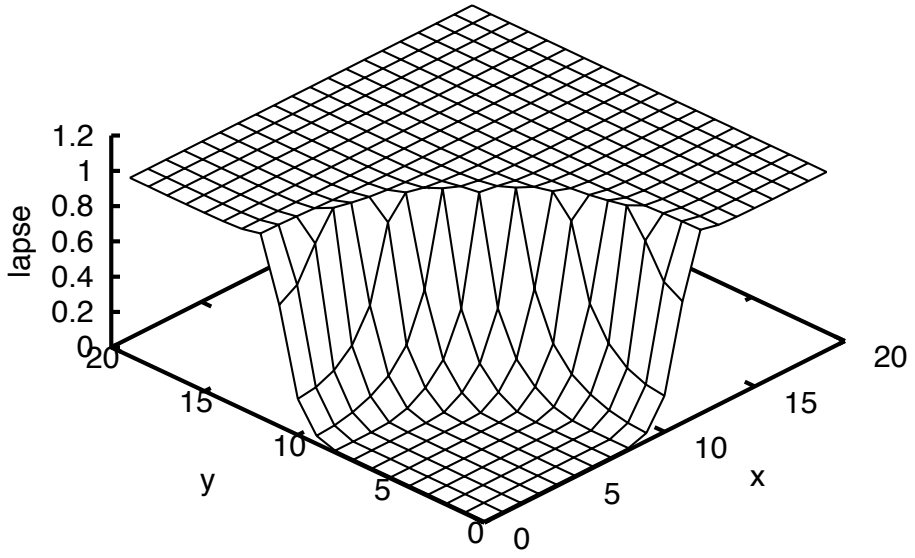


FIGURE 3.3: Plot of the lapse function for a single BH at $t = 1000\,M$ in normal coordinates. Only one of every ten points is shown along each direction. The third-order accurate algorithm has been used with $\beta = 1/12$ and a space resolution $h = 0.1\,M$. The profile is steep, but smooth: no sign of instability appears. Small riddles, barely visible on the top of the collapse front, signal some lack of resolution because of the logarithmic character of the grid. The dynamical zone is safely away from the boundaries.

3.5 Single Black hole test: first-order shift conditions

Looking at the results of the previous section, one can wonder wether our code is just tuned for normal coordinates. This is why we will consider here again BH simulations, but this time with some non-trivial shift prescriptions. The idea is just to test some simple cases in order to show the gauge-polyvalence of the code. For the sake of simplicity, we will consider here just first order shift prescriptions, meaning that the source

terms (Q, Q^i) in the gauge evolutions (3.7) are algebraic combinations of the remaining dynamical fields. To be more specific, we shall keep considering slicing conditions defined by

$$Q = -\beta^k/\alpha\, A_k + f\ (tr\, K - m\, \Theta)\ , \tag{3.33}$$

together with dynamical shift prescriptions, defined by different choices of Q^i.

First-order shift prescriptions have been yet considered at the theoretical level [29]. We will introduce here an additional requirement, which follows when realizing that, allowing for the 3+1 decomposition of the line element

$$ds^2 = -\alpha^2\ dt^2 + \gamma_{ij}\,(dx^i + \beta^i dt)\,(dx^j + \beta^j dt)\ , \tag{3.34}$$

the shift behaves as a vector under (time independent) transformations of the space coordinates. We will impose then that its evolution equation, and then Q^i, is also three-covariant.

This three-covariance requirement could seem a trivial one. But note that the harmonic shift conditions, derived from

$$\Box\ x^i = 0, \tag{3.35}$$

are not three-covariant (the box here stands for the wave operator acting on scalars). In the 3+1 language, (3.35) can be translated as

$$\partial_t(\sqrt{\gamma}/\alpha\ \beta^i) - \partial_k(\sqrt{\gamma}/\alpha\ \beta^k\beta^i) + \partial_k(\alpha\sqrt{\gamma}\ \gamma^{ik}) = 0\ , \tag{3.36}$$

where the non-covariance comes from the space-derivatives terms.

Concerning the advection term, a three-covariant alternative would be provided either by the Lie-derivative term

$$\mathcal{L}_\beta\,(\,\sqrt{\gamma}\ \beta^i/\alpha\,) = \mathcal{L}_\beta\,(\,\sqrt{\gamma}/\alpha\,)\ \beta^i\ , \tag{3.37}$$

or by the three-covariant derivative term

$$\beta^k\,\nabla_k(\,\beta^i/\alpha\,) = 1/\alpha\,[\,\beta^k\,B_k{}^i - \beta^i\beta^k A_k + \Gamma^i_{j\,k}\,\beta^j\beta^k\,]\ . \tag{3.38}$$

We have tested both cases in our numerical simulations.

Concerning the last term in (3.36), we can take any combination of A^i, Z^i and the vectors obtained form the space metric derivatives after subtracting their initial values, namely:

$$D_i - D_i\,|_{t=0}\ , \qquad E_i - E_i\,|_{t=0}\ . \tag{3.39}$$

This is because the additional terms arising in the transformation of the non-covariant quantities (D_i, E_i) depend only on the space coordinates transformation, which is assumed to be time-independent. Note that, for the conformal contracted-Gamma combination

$$\Gamma_i = 2\,E_i - \frac{2}{3}\,D_i \ , \tag{3.40}$$

the subtracted terms actually vanish in simulations starting from the isotropic initial metric (3.31). Of course, the same remark applies to the BSSN Gamma quantity, namely [19]

$$\widetilde{\Gamma}_i = \Gamma_i + 2\,Z_i \ . \tag{3.41}$$

We have considered the following combinations:

$$S1: \qquad \partial_t\,\beta^i = \frac{\alpha^2}{2}\,A^i - \alpha\,Q\,\beta^i \tag{3.42}$$

$$S2: \qquad \partial_t\,\beta^i = \frac{\alpha^2}{2}\,A^i + \beta^k\,B_k{}^i + \Gamma^i_{j\,k}\,\beta^j\beta^k - \alpha\,Q\,\beta^i \tag{3.43}$$

$$S3: \qquad \partial_t\,\beta^i = \frac{\alpha^2}{4}\,\widetilde{\Gamma}^i + \beta^k\,B_k{}^i + \Gamma^i_{j\,k}\,\beta^j\beta^k - \alpha\,Q\,\beta^i \ , \tag{3.44}$$

where S1 corresponds to the Lie-derivative term (3.37) and the remaining two choices to the covariant advection term (3.38), with different combinations of the first-order vector fields.

We have obtained stable evolution in all cases, with the simulations lasting up to the point when the collapse front crosses the outer boundary (about $t = 50\,M$). We can see in Fig. 3.4 the lapse and shift profiles in the S1 and the S3 cases (S2 is very similar to S1). The shift profiles are modulated by the lapse ones, so that the shift goes to zero in the collapsed regions. This is a consequence of the term $-\alpha\,Q\,\beta^i$ in the shift evolution equation, devised for getting finite values of the combination β^i/α. In the non-collapsed region, S1 leads to a higher shift profile, which spreads out with time, whereas S3 leads to a lower profile, which starts diminishing after the initial growing. Allowing for (3.44), this indicates that the conformal gamma quantity $\widetilde{\Gamma}_i$ is driven to zero. The lapse slopes are also slightly softened in the S3 case.

These results confirm that the code stability is not linked to any particular shift prescription, as we can combine different source terms in the shift evolution equation, leading to different lapse and shift profiles.

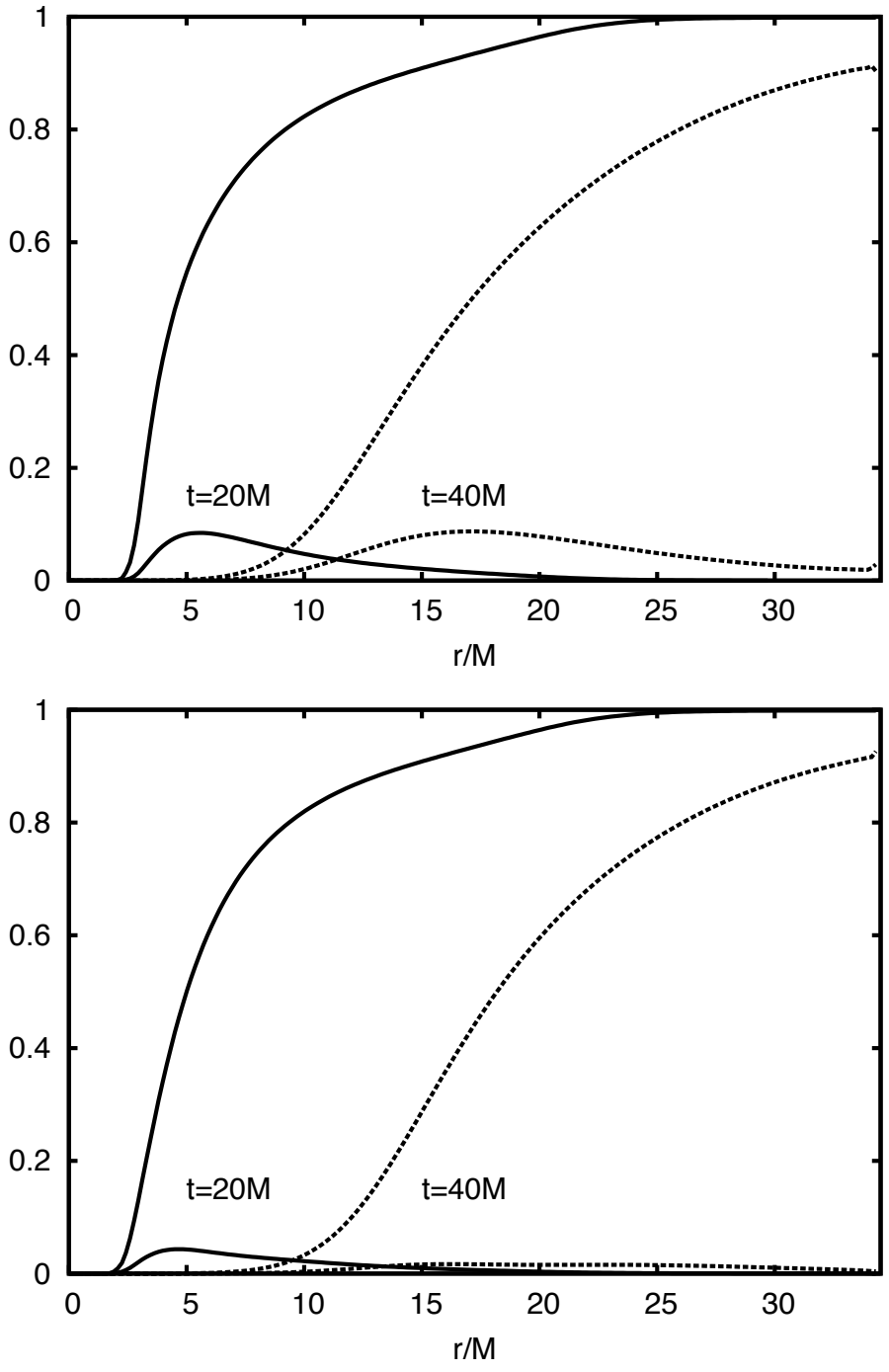


FIGURE 3.4: Plot of the lapse and shift profiles at $t = 20\,M$ (continuous lines) and $t = 40\,M$ (dotted lines). The plots are shown along the main diagonal of the computational domain, in order to keep the outer boundary out of the dynamical zone. In the S1 case (up panel), after the initial growing, the maximum shift value keeps constant. In the S3 case (down panel), it clearly diminishes with time.

3.6 Summary

We have therefore shown in this chapter how a first-order flux-conservative version of the Z4 formalism can be adjusted for dealing with the ordering constraints, and then implemented in a numerical code by means of a robust, cost-efficient, finite-difference formula. The resulting scheme has been tested in a demanding harmonic-coordinates scenario: the gauge-waves testbed. The code performance compares well with the best harmonic-code results for this test [11], even in the highly non-linear regime (50% amplitude case). This is in contrast with the well-known problems of BSSN-based codes with the gauge-waves test [1] [8].

The code has also been tested in non-excision BH evolutions, where singularity-avoidance is a requirement. Our results confirm the robustness of the code for many different choices of dynamical lapse and shift prescriptions. In the normal coordinates case (zero shift), our results set up a new benchmark, by evolving the BH up to $1000\,M$ without any sign of instability. This improves the reported BSSN result by one order of magnitude (Harmonic codes are not devised for normal coordinates). More important, this shows that a specific shift choice is not crucial for code stability, even in non-excision BH simulations. This is confirmed by our shift simulations, where different covariant evolution equations for the shift lead also to stable numerical evolution.

References

[1] K. Kiuchi and H-A. Shinkai, Phys. Rev. D**77**, 044010 (2008).

[2] G. Yoneda and H-A. Shinkai, Phys. Rev. D**66**, 124003 (2002).

[3] M. Shibata and Y.-I. Sekiguchi, Phys. Rev. D**72**, 044014 (2005).

[4] M. D. Duez, Y. T. Liu, S. L. Shapiro, and B. C. Stephens, Phys. Rev. D**72**, 024028 (2005).

[5] M. Anderson, E. W. Hirschmann, S. L. Liebling, and D. Neilsen, Class. Quantum Grav. **23**, 6503 (2006).

[6] B. Giacomazzo and L. Rezzolla, Class. Quantum Grav. **24**, 235 (2007).

[7] M. Alcubierre et al, Phys. Rev. D**67**, 084023 (2003).

[8] M. Alcubierre et al, Class. Quantum Grav. **21**, 589 (2004).

[9] N. Jansen, B. Bruegmann and W. Tichy, Phys. Rev. D**74**, 084022 (2006).

[10] Y. Zlochower, J. G. Baker, M. Campanelli and C. O. Lousto, Phys. Rev. D**72**, 024021 (2005).

[11] M. Babiuc et al, Class. Quant. Grav. **25**, 125012 (2008).

[9] C. Bona, T. Ledvinka, C. Palenzuela, M. Žáček, Phys. Rev. D **67**, 104005 (2003).

[14] C. Gundlach, G. Calabrese, I. Hinder and J.M. Martín-García, Class. Quantum Grav. **22**, 3767 (2005).

[14] C. Bona, C. Bona-Casas and J. Terradas, J. Comp. Physics **228**, 2266 (2009).

[15] S. Osher and S. Chakravarthy, ICASE Report **84–44**,
ICASE NASA Langley Research Center, Hampton, VA (1984).

[16] D. Alic, C. Bona, C. Bona-Casas and J. Massó, Phys. Rev. D**76**, 104007 (2007)

[17] C. Bona, J. Massó, E. Seidel and J. Stela, Phys. Rev. Lett. **75** 600 (1995).

[18] M. Holst, et al, Phys. Rev. **D70** 084017 (2004).

[19] C. Bona, T. Ledvinka, C. Palenzuela and M. Žáček, Phys. Rev. D **69**, 064036 (2004).

[20] G. Nagy, O. Ortiz and O. Reula, Phys. Rev. **D70** 044012 (2004).

[21] C. Bona and J. Massó, Phys. Rev. Lett. **68**, 1097 (1992).

[22] www.appleswithapples.org/TestResults/Results/Abigel05/Abigel05.html

[23] M. Tiglio, L. Lehner and D. Neilsen, Phys. Rev. **D70** 104018 (2004).

[24] L. E. Kidder, M. A. Scheel and S. A. Teukolsky, Phys. Rev. **D64** 064017 (2001).

[25] D. Bernstein, *Ph. D. Thesis*, Dept. of Physics, Univ. of Illinois at Urbana-Champaign (1993).

[26] A. Arbona et al., Phys. Rev. **D57** 2397 (1998).

[27] A. Arbona, C. Bona, J. Massó and J. Stela, Phys. Rev. **D60** 104014 (1999).

[28] B. Reimann and B. Bruegmann, Phys. Rev. D**69**, 044006 (2004).

[29] C. Bona and C. Palenzuela, Phys. Rev. **D69** 104003 (2004).

Chapter 4

Constraint-preserving boundary conditions

4.1 Introduction

Constraint-preserving boundary conditions is an active research topic in Numerical Relativity. During the first half of this decade, many conditions have been proposed, adapted in each case to some specific 3+1 evolution formalism: Fritelli-Reula [1], KST [2, 3], BSSN-NOR [4], or Z4 [5]. The focus changed suddenly after 2005 by the impact of a breakthrough: the first 'long term' binary-black-hole simulation, which was achieved in a generalized-harmonic formalism [6]. A series of constraint-preserving boundary conditions proposals in this framework started then [7, 8], and continues today [9–11].

We will retake in this chapter the 3+1 approach to constraint-preserving boundary conditions, following the way opened very recently for the BSSN case [12]. More specifically, we will revisit the Z4 case, not just because of its intrinsic relevance, but also for its relationship with other 3+1 formulations (BSSN, KST, see refs. [12, 13] for details). Also, the close relationship between the Z4 and the generalized-harmonic formulations suggest that our results could provide a different perspective in this other context. This was actually what happened with the current constraint-damping terms: first derived in the Z4 context [14] and then applied successfully in generalized-harmonic simulations [6].

Our results are both at the theoretical and the numerical level. We will consider the first-order Z4 formalism in normal coordinates (zero shift) for the harmonic slicing case. This case was known to be symmetric-hyperbolic for a particular choice of the parameter which controls the ordering of space derivatives [9, 15]. We will extend this result to a range of this ordering parameter, by providing explicitly a positive-definite energy

estimate. Then we will use this estimate for deriving algebraic constraint-preserving boundary conditions both for the energy and the normal momentum components.

Afterwards we will consider the dynamical evolution of constraint violations (subsidiary system). Following standard methods [2–4], we will transform algebraic boundary conditions of the subsidiary system into derivative boundary conditions for the main system. We will introduce a new basis of dynamical fields in order to revise the constraint-preserving conditions proposed in refs. [5, 15] for the Z4 formalism, including also a new coupling parameter which affects the propagation speeds of the (modified) incoming modes. In the case of the energy constraint, we get a closed subsystem for the principal part, allowing an analytical stability study at the continuum level which is presented in Appendix G.

A simple numerical implementation of the proposed conditions will be given, where we will test the stability in the linear regime, by considering small random-noise perturbations around flat space (robust stability test). The results show the numerical stability of the proposed boundary conditions in this regime for many different combinations of the parameters. The space discretization scheme is the simplest one with the summation-by-parts (SBP) property [16]. In this way we avoid masking the effect of our conditions (at the continuum level) with the effect of more advanced space-discretization algorithms, like FDOC (see Chapter 2 or [17]) devised to reduce the high frequency noise level in long-term simulations, which has recently been applied to the black-hole case [18] as we have seen in the previous chapter. For a comparison, we will run also with periodic boundary conditions, where the noise level keeps constant. The proposed boundary conditions produce instead a very effective decreasing of (the cumulated effect of) energy and momentum constraint violations. In the case of cartesian-like grids, we also compare the standard 'a la Olsson' treatment [16], with a modified numerical implementation which does not use the corner and vertex points, avoiding in this way some stability issues and providing much cleaner evidence of constraint preservation.

Finally we will test the non-linear regime with the Gowdy waves [19] metric, one of the standard numerical relativity code tests, as we have done recently for the energy constraint case [20]. We endorse in this way some recent claims (by Winicour and others) that the current code cross-comparison efforts [8, 11] should be extended to the boundaries treatment. A convergence test will be performed against this exact strong-field solution, showing the expected convergence rate (second order for our simple SBP method). Testing the proposed boundary conditions results into a stable and constraint-preserving behavior, in the sense that energy and momentum constraint violations remain similar or even smaller than the corresponding effects with exact (periodic or reflection) boundary conditions for the Gowdy metric.

4.2 The Z4 case revisited

We will consider again the Z4 evolution system:

$$R_{\mu\nu} + \nabla_\mu Z_\nu + \nabla_\nu Z_\mu = 8\,\pi\,(T_{\mu\nu} - \frac{1}{2}\,T\,g_{\mu\nu}). \tag{4.1}$$

More specifically, we will consider the first-order version in normal coordinates, as described in refs. [12, 15] and in previous chapter. For further convenience, we will recombine the basic first-order fields (K_{ij}, D_{ijk}, A_i, Θ, Z_i) in the following way:

$$\begin{aligned}
\Pi_{ij} &= K_{ij} - (\,trK - \Theta)\,\gamma_{ij} && (4.2)\\
V_i &= \gamma^{rs}(D_{irs} - D_{ris}) - Z_k && (4.3)\\
\mu_{ijk} &= D_{ijk} - (\gamma^{rs}D_{irs} - V_i)\,\gamma_{jk} && (4.4)\\
W_i &= A_i - \gamma^{rs}D_{irs} + 2\,V_i && (4.5)
\end{aligned}$$

so that the new basis is (Π_{ij}, μ_{ijk}, W_i, Θ, V_i). Note that the vector Z_i can be recovered easily as

$$Z_i = -\mu^k{}_{ik}. \tag{4.6}$$

With this new choice of basic dynamical fields, the principal part of the evolution system gets a very simple form in the harmonic slicing case:

$$\begin{aligned}
\partial_t\,W_i &= \cdots && (4.7)\\
\partial_t\,\Theta + \alpha\;\partial_k\,V^k &= \cdots && (4.8)\\
\partial_t\,V_i + \alpha\;\partial_i\,\Theta &= \cdots && (4.9)\\
\partial_t\,\Pi_{ij} + \alpha\;\partial_k\,\lambda^k{}_{ij} &= \cdots && (4.10)\\
\partial_t\,\mu_{kij} + \alpha\;\partial_k\,\Pi_{ij} &= \cdots && (4.11)
\end{aligned}$$

where the dots stand for non-principal contributions, and we have noted for short

$$\lambda_{kij} = \mu_{kij} + \gamma_{k(i}W_{j)} - W_k\,\gamma_{ij} - (1+\zeta)\,[\,\mu_{(ij)k} + \gamma_{k(i}Z_{j)}\,]\,, \tag{4.12}$$

where ζ is a space-derivatives ordering parameter and round brackets denote index symmetrization.

The first-order version of the Z4 system is known to be symmetric-hyperbolic in normal coordinates with harmonic slicing, at least for the usual ordering $\zeta = -1$ [21]. It follows from (4.7-4.11) that this result can be extended to the following range of the ordering

parameter

$$-1 \leq \zeta \leq 0\,, \tag{4.13}$$

which covers the symmetric ordering case ($\zeta = 0$). The corresponding 'symmetrizer', or 'energy estimate', can be written as:

$$S = \Theta^2 + V_k V^k + \Pi^{ij}\,\Pi_{ij} + \tilde{\mu}^{kij}\tilde{\mu}_{kij} + (1+\zeta)(Z^k Z_k - \tilde{\mu}^{kij}\tilde{\mu}_{ijk}) + 2\,\zeta\, Z_k W^k\,, \tag{4.14}$$

where we have noted for short

$$\tilde{\mu}_{kij} = \mu_{kij} - W_k\,\gamma_{ij}\,. \tag{4.15}$$

Allowing for (4.7-4.11), we get

$$\frac{-1}{2\,\alpha}\,\partial_t\, S = \partial_k\,(\Theta\, V^k + \Pi_{ij}\,\lambda^{kij}) + \cdots \tag{4.16}$$

and the divergence theorem can be used in order to complete the proof. The positivity proof for S for the interval (4.13) is given in Appendix F.

Characteristic decomposition

We can consider now some specific space surface, in order to identify the constraint modes by looking at the evolution equations for Θ and Z_i in the system (4.7-4.11). It follows from (4.8, 4.9) that the energy-constraint modes are given by the pair

$$E^{\pm} = \Theta \;\pm\; V_n \tag{4.17}$$

with propagation speed $\pm\alpha$ (the index n meaning the projection along the unit normal n_i). Also, allowing for (4.6,4.11), we can easily recover the evolution equation for Z_i, namely

$$\partial_t\, Z_i - \alpha\;\partial_k\,\Pi^k{}_i = \cdots \tag{4.18}$$

so that we can identify the momentum-constraint modes with the three pairs, with propagation speed $\pm\alpha$,

$$M_i^{\pm} = \Pi_{ni} \;\pm\; \lambda_{nni}\;. \tag{4.19}$$

Note that, allowing for (4.2), the normal component Π_{nn} does correspond with the transverse-trace component of the extrinsic curvature K_{ij}. We give for completeness the remaining modes, the fully tangent ones, with propagation speed $\pm\alpha$,

$$T_{AB}^{\pm} = \Pi_{AB} \;\pm\; \lambda_{nAB} \tag{4.20}$$

(capital indices denote a projection tangent to the surface), and the standing modes (zero propagation speed):

$$W_i \ , \qquad V_A \ , \qquad \mu_{Aij} \ . \tag{4.21}$$

Algebraic boundary conditions

We can take advantage of the positive-definite energy estimate (4.14) in order to derive suitable algebraic boundary conditions. We can integrate (4.16) in space and, by applying the divergence theorem, we get a positivity condition for the boundary terms, namely

$$(\Pi^{ij} \ \lambda_{nij} + \Theta \ V_n) \ |_\Sigma \ \ \geq \ 0 \tag{4.22}$$

where Σ stands for the boundary surface ($\mathbf{n}$ being here its outward normal).

The contribution of the fully tangent modes (4.20), independent of the energy and momentum sectors, is given by

$$\Pi^{AB} \ \lambda_{nAB} = \frac{1}{4} \ tr \left[(T^+)^2 - (T^-)^2 \right] , \tag{4.23}$$

so that the contribution of these modes to the boundary term in (4.22) will be non-negative if we impose the standard algebraic boundary-conditions:

$$T^-_{AB} = \sigma \, T^+_{AB} \qquad |\sigma| \leq 1 \, , \tag{4.24}$$

the case $\sigma = 0$ corresponding to maximal dissipation. A less strict condition is obtained by adding an inhomogeneous term, namely

$$T^-_{AB} = \sigma \, T^+_{AB} + G_{AB} \, . \tag{4.25}$$

This can cause some growth of the energy estimate but, provided that the array G consists of prescribed spacetime functions, the growth rate can be bounded in a suitable way so that a well-posed system can still be obtained (see for instance refs. [4, 12]).

This simple strategy, when applied to the energy and momentum modes (4.17, 4.19) is not compatible with constraint preservation in the generic case (see also ref. [3]). For the energy sector, constraint preservation is obtained only for the extreme case:

$$E^- = E^+ \qquad \Leftrightarrow \qquad \Theta = 0 \, , \tag{4.26}$$

which will reflect energy-constraint violations back into the evolution domain. These conditions would be then of a limited practical use in realistic simulations.

A different approach can be obtained by realizing that the contribution to the boundary term in (4.22) would have the right sign if one uses the following 'logical gate' condition:

$$\Theta\,|_{\Sigma} \;= 0 \qquad if \qquad (\Theta\; V_n)\,|_{\Sigma} \;< \;0 \tag{4.27}$$

(Θ-gate in ref. [20]). It is clear that the boundary condition (4.27) preserves the energy constraint, as it modifies just the Θ values, by setting them to zero when the condition is fulfilled, without affecting any other dynamical field.

The same strategy can work for normal components of the momentum modes (4.19), at least for the symmetric choice of the ordering parameter. Allowing for (4.12), one has

$$M_n^{\pm} = \Pi_{nn} \mp Z_n \qquad (\zeta = 0)\,, \tag{4.28}$$

so that a constraint-preserving (reflection) condition can be obtained in the extreme case as well. In the logical gate approach, the contribution of the modes (4.28) to the boundary term in (4.22) will have the right sign if one uses the condition (case $\zeta = 0$ only):

$$Z_n\,|_{\Sigma} \;= 0 \qquad if \qquad (Z_n\;\Pi_{nn})\,|_{\Sigma} \;> \;0\,, \tag{4.29}$$

which clearly preserves the normal component of the momentum constraint.

For the tangent momentum modes $M_A^{\pm}$ (tangent to the boundary surface), however, the contribution in (4.22) will be

$$2\,\Pi^{nA}\;\lambda_{nnA}\,, \tag{4.30}$$

where λ_{nnA} is inhomogeneous in Z_A for any value of the ordering parameter. Moreover, the inhomogeneous terms are not prescribed functions, but rather some combinations of dynamical fields. A different strategy must then be devised in this case, as we will see below.

4.3 Constraints evolution and derivative boundary conditions

The time evolution of the energy-momentum constraints can be easily derived by taking the divergence of the Z4 field equations (5.4), that is

$$\Box\; Z_\mu + R_{\mu\nu} Z^\nu = 0\;. \tag{4.31}$$

We can write down the second order equation (4.31) as a first order system and impose then maximally dissipative boundary conditions on (the first derivatives of) the

Z_μ components. In this way, the boundaries will behave as one-way membranes for constraint-violating modes, at least for the ones propagating along the normal direction n_i.

The procedure can be illustrated with the energy-constraint, that is the time component of (4.31):

$$\partial^2_{tt}\,\Theta - \alpha^2 \triangle \Theta = \cdots \tag{4.32}$$

A first-order version can be obtained as usual by considering first-order derivatives as independent quantities, namely

$$\dot{\Theta} \equiv 1/\alpha\ \partial_t\,\Theta\,, \qquad \Theta_k \equiv \partial_k\,\Theta\,. \tag{4.33}$$

We can write then (4.32) as the following first-order symmetric-hyperbolic system

$$\partial_t\ \dot{\Theta} - \alpha\ \partial_k\,\Theta^k \;=\; \cdots \tag{4.34}$$

$$\partial_t\ \Theta_k - \alpha\ \partial_k\,\dot{\Theta} \;=\; \cdots \tag{4.35}$$

Boundary conditions for (the incoming modes of) the subsidiary system can be enforced then in the standard way. We will consider here for simplicity the 'maximal dissipation' condition, that is (we assume that the boundary is on the right):

$$\dot{\Theta} \;+\; n^k\Theta_k = 0\,. \tag{4.36}$$

Now we can use it as a tool for setting up boundary conditions for the energy modes of the main evolution system. One can for instance enforce directly (4.36), as in ref. [20].

We will rather use (4.36) as a tool for getting (derivative) boundary conditions for the incoming energy mode of the evolution system (4.7 - 4.11). To do this, we can use the evolution equation (4.8) for transforming (4.36) into a convenient version of the energy constraint, namely:

$$\mathcal{E} \;=\; \partial_k\,V^k - \partial_n\,\Theta + \cdots \tag{4.37}$$

We can now use (4.37) in order to modify the evolution equation of the incoming energy mode E^-, that is:

$$1/\alpha\ \partial_t\ E^- + \partial_k\,V^k - \partial_n\,\Theta \;=\; a\ \mathcal{E} + \cdots \tag{4.38}$$

The whole process is equivalent to the simple replacement:

$$E^- \;\rightarrow\; E^- + a\ (\Theta^{(adv)} - \Theta) \tag{4.39}$$

where $\Theta^{(adv)}$ is the solution of the advection equation (4.36).

The choice $a = 1$ corresponds to the standard recipe [2–4] of 'trading' space normal derivatives by time derivatives, in the incoming modes. This implies that the modified mode gets zero propagation speed along the given direction $\mathbf{n}$. In this case, allowing for (4.38), the time derivative of E^- would actually vanish, modulo non-principal terms; this amounts to freezing the incoming modes to their initial values (maximal dissipation 'on the right-hand-side'), which is a current practice in some Numerical Relativity codes. Note however that constraint preservation requires using the right non-principal terms, that can be deduced from the full expression (4.39).

The choice $a = 2$ would imply instead that the modified mode gets the same positive speed $(+\alpha)$ than the outgoing one E^+. We show in Appendix G that this choice will lead to a weakly-hyperbolic (ill-posed) boundary system. Our results confirm that $a = 1$ is actually a safe choice [2–4], although other values in the interval $1 \leq a < 2$ lead also to a strongly hyperbolic system with non-negative speeds for all energy modes (see Appendix G for details).

Momentum constraint conditions

The same method can be applied to the momentum constraint modes, although in a less straightforward way. Let us start from the evolution equation (4.18) for Z_i, and take one extra time derivative. We get in this way

$$\partial^2_{tt}\, Z_i + \alpha^2\, \partial^2_{rs}\, \lambda^{rs}{}_i \;\; = \;\; \cdots \tag{4.40}$$

which, after some cross-derivatives cancellations, leads to the space components of (the principal part of) the covariant equation (4.31).

A first-order version of (4.40) can be obtained again by considering first-order derivatives as independent quantities. For the time derivative we will take the obvious choice

$$\dot{Z}_i \equiv 1/\alpha\; \partial_t\, Z_i\,. \tag{4.41}$$

The treatment of space derivatives, however, is complicated by the fact that we are dealing with a first-order formulation, so that there are additional ordering constraints to be allowed for. Following refs. [5, 15], we will define for further convenience

$$Z_{ki} \equiv \partial_k\, Z_i - \partial_{[k}\, A_{i]} - \partial_{[k}\, D_{i]} - (1-\zeta)\, \gamma^{rs}\, \partial_{[r}\, D_{k]\,is} + (1+\zeta)\, \gamma^{rs}\, \partial_{[r}\, D_{i]\,ks}\,, \tag{4.42}$$

where we have noted for short $D_i = \gamma^{rs} D_{irs}$. A closer look to (4.42) shows that Z_{ki} is just the space derivative of Z_i, modulo ordering constraints. In the notation of this chapter:

$$Z_{ki} = -\partial_r \, \mu_{ki}{}^r + \partial_{[i} \, W_{k]} + (1+\zeta) \, [\, \partial_r \, \mu_{(ki)}{}^r + \partial_{(k} \, Z_{i)} \,] + \cdots \tag{4.43}$$

We can write now (4.40) in the first-order form

$$\partial_t \, \dot{Z}_i - \alpha \, \partial_k \, Z^k{}_i \;\; = \;\; \cdots \tag{4.44}$$

$$\partial_t \, Z_{ki} - \alpha \, \partial_k \, \dot{Z}_i \;\; = \;\; \cdots \tag{4.45}$$

which is a symmetric-hyperbolic first-order version of the momentum-constraint evolution system (other versions could be obtained by playing with the ordering constraints in a different way). The vanishing of the incoming modes of this subsidiary system can be enforced now in the same way as for the energy constraint, namely:

$$\dot{Z}_i \;\; + \;\; n^k Z_{ki} = 0 \, . \tag{4.46}$$

This is obviously a 'maximal dissipation' constraint-preserving condition for the subsidiary system, which can be used for to get a derivative boundary condition for the main evolution system, as we did for the energy modes in the preceding subsection. To be more specific, we can use the evolution equation (4.18) for transforming (4.46) into a convenient version of the momentum constraint, that is

$$\mathcal{M}_i \;\; = \;\; \partial_k \, \Pi^k{}_i + n^k Z_{ki} + \cdots \tag{4.47}$$

and use it for modifying the evolution equation of the incoming momentum modes M_i^-, namely:

$$1/\alpha \; \partial_t \; M_i^- + \partial_k \, \lambda^k{}_{ni} - \partial_n \, \Pi_{ni} \;\; = \;\; -a \; \mathcal{M}_i + \cdots \tag{4.48}$$

which amounts to the following replacement:

$$M_i^- \;\; \rightarrow \;\; M_i^- + a \; (Z_i^{(adv)} - Z_i) \, , \tag{4.49}$$

where $Z_i^{(adv)}$ is the solution of the advection-like equation (4.46).

The choice $a = 1$ would imply again that the modified modes get zero propagation speeds along the normal direction, whereas the choice $a = 2$ would imply instead that the modified modes get the same positive speed $(+\alpha)$ than the outgoing ones M_i^+. This result requires the extra ordering terms in (4.42): this was actually the reason for including them. Note that we can consider different values of the coupling parameter

for the energy modes ($a = a_E$), and even for the normal and tangent momentum modes ($a = a_N$, a_T , respectively).

For any value $a \geq 1$, the modified modes can be computed consistently from inside. The momentum system however is too complicated for a full hyperbolicity analysis, like the one we provide in Appendix G for the energy sector. Part of the complication comes from the coupling with the non-constraint modes, which require their own boundary conditions. Let us remember at this point that the boundary conditions presented in this section are derivative, not algebraic. This means that, even in the symmetric hyperbolic cases, proving well-posedness is by no means trivial.

For that reason, we will rather follow the approach of ref. [3], focusing in the stability of small perturbations around smooth solutions, which can be tested numerically. We start in the following section, by performing a 'robust stability' test in order to check the numerical stability of high-frequency perturbations around the Minkowsky metric. As a full set of boundary conditions is required, even in this weak-field test, we supplement our conditions for the constraint-related modes with the freezing of the initial values of the incoming non-constraint modes (maximal dissipation 'on the right-hand-side').

4.4 Numerical implementation

Let us test now the stability and performance of the proposed conditions in the linear weak-field regime, by considering a small perturbation of Minkowski space-time, which is generated by taking random data both in the extrinsic curvature and in the constraint-violation quantities (Θ, Z_i). In this way the initial data violate the energy-momentum constraints, but preserve all ordering constraints. The level of the random noise will be of the order 10^{-6}, small enough to make sure that we will keep in the linear regime during the whole simulation (Robust Stability test, see ref. [23] for details).

We will use the standard method of lines [24] as a finite difference algorithm, so that space and time discretization will be treated separately. The time evolution will be dealt with a third-order Runge-Kutta algorithm. The time step dt is kept small enough to avoid an excess of numerical dissipation that could distort our results in long runs.

For space discretization, we will consider a three-dimensional rectangular grid, evenly-spaced along every space direction, with a space resolution $h = 1/80$. We will use there the simplest centered, second-order-accurate, discretization scheme. At the points next to the boundary, where we can not use the required three-points stencil, we will switch to the standard first-order upwind (outgoing) scheme. This combination is the simplest one with the summation-by-parts (SBP) property [16]. In this way we expect

that the theoretical properties derived from symmetric-hyperbolicity will show up in the simulations in a more transparent way. For the same reason, we avoid adding extra viscosity terms that could mask the effect of our conditions (at the continuum level) with the dissipative effects of the discretization algorithm. Just to make sure, we run also with periodic boundary conditions, where the noise level keeps constant: any decrease of the constraint-violation level will then be due to the proposed conditions, not to the discretization scheme.

Let us be more specific about the boundary treatment. At boundary points, we use the first-order upwind algorithm in order to get a prediction for every dynamical field. Once we have got this prediction, we perform the characteristic decomposition along the direction normal to the boundary. The predicted values for the outgoing modes, for which the upwind algorithm is known to be stable, will be kept (this includes the 'standing' modes, with zero characteristic speed). The (unstable) incoming modes will be replaced instead by the values arising from our boundary conditions, as described in the preceding section.

We start with simulations in which the proposed conditions are applied just to the z face, whereas we keep periodic boundary conditions along the x and y directions. In this way

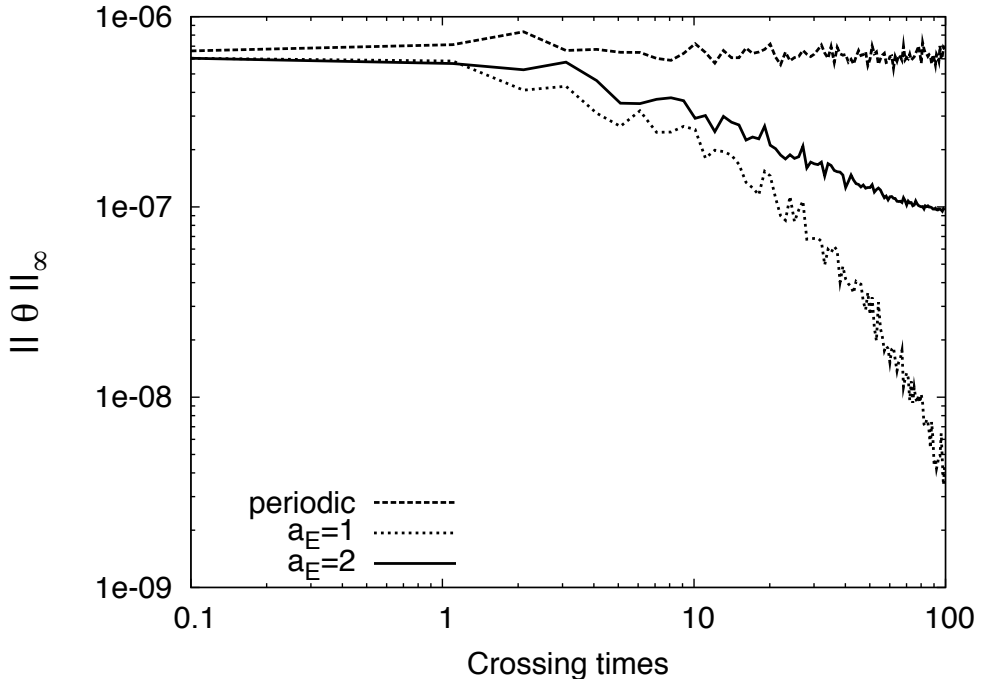


FIGURE 4.1: Robust stability test. Time evolution of the maximum norm of Θ, with just one face open (periodic boundaries are implemented along the transverse directions). The fully periodic boundaries result (dashed lines) is also included for comparison. We see some growing mode onset in the $a_E = 2$ case, whereas the constraint-preserving $a_E = 1$ case (continuous line) is very efficient at reducing the initial noise level.

we can detect instabilities which are inherent to the proposed boundary conditions on smooth boundaries (no corners), allowing at the same time for some non-trivial dynamics along at least one tangent direction (because of the rectangular nature of the grid).

We plot in fig. 4.1 the maximum norm of the energy-constraint-violating quantity Θ for two different choices of the coupling parameter of the energy mode: $a_E = 1\,, 2\,.$ We can see that, after 100 crossing times, the case $a_E = 2$ starts showing the effect of the linear modes predicted by our hyperbolicity analysis in Appendix G, by departing from the maximal dissipation pattern of decay. We plot for comparison the results obtained by applying periodic boundary conditions, so we can see how, for the choice $a_E = 1$, the proposed constraint-preserving conditions are extremely effective at 'draining out' energy constraint violations. The rate of decay is actually the same as the one obtained by applying maximal dissipation conditions 'on the right-hand-side' also to the energy modes, as expected from the analysis given in the previous section. In what follows, we will fix $a_E = 1$ for this coupling parameter.

We plot in fig. 4.2 both the maximum norm of the longitudinal (upper panel) and transverse components (lower panel) of the momentum-constraint-violating vector Z_i for the choice $a_N = a_T = 1$ of the coupling parameter of the momentum modes. We include again for comparison the results obtained by applying periodic boundary conditions, so we can see how the proposed constraint-preserving conditions are very effective at 'draining out' energy constraint violations. The Z_y plots are slightly, sensitive to the ordering parameter $\zeta = 0, -1$. In the $\zeta = -1$ case, the rate of decay is actually the same as the one obtained by applying instead maximal dissipation conditions 'on the right-hand.-side' for the momentum modes. The results are qualitatively the same for other components of Z_i and for other parameter combinations $a_N, a_T = 1, 2\,.$

In order to perform a full test for cartesian-like grids, including corner and vertex points, we will repeat the same simulations, but this time with the proposed boundary conditions applied to all faces, not just to the z ones. A standard treatment of corner points 'a la Olsson' [16], like the one presented in previous works [5, 15], results into numerical instability issues. A simple cure is to add some extra dissipation at the interior points, at the price of masking the theoretical results, at the continuum level, with the numerical viscosity effects, as shown in fig. 4.3. We can see there that opening all faces makes the effects to appear much faster. The expected instability of the $a_E = 2$ choice of the energy coupling parameter, which was just an onset in fig. 4.1, shows up manifestly here (upper panel). Also, a growing mode onset is clearly visible for the choice $\zeta = -1$ of the momentum-constraint coupling parameter (lower panel). The case $\zeta = 0$ looks stable, although no strong conclusion can be drawn because of the added numerical dissipation.

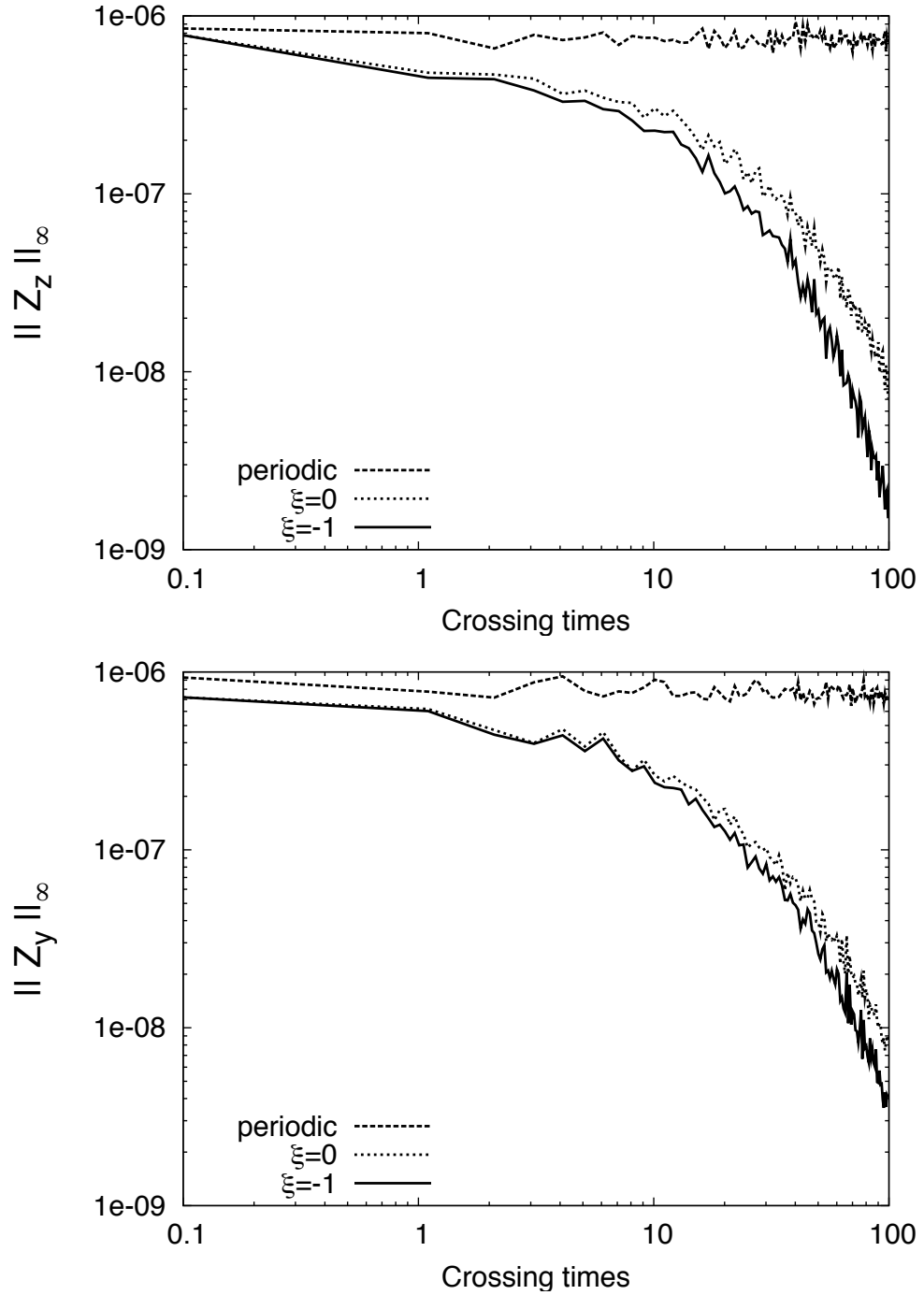


FIGURE 4.2: Robust stability test. Time evolution of the maximum norm of the longitudinal and transverse Z_i components (up and down panels, respectively). In both cases, the periodic boundaries results (dashed lines) are included for comparison. The initial noise in the momentum constraint gets reduced very efficiently in both the $\zeta = -1$ and the $\zeta = 0$ cases, although there is a slight difference, more visible in the longitudinal case (upper panel).

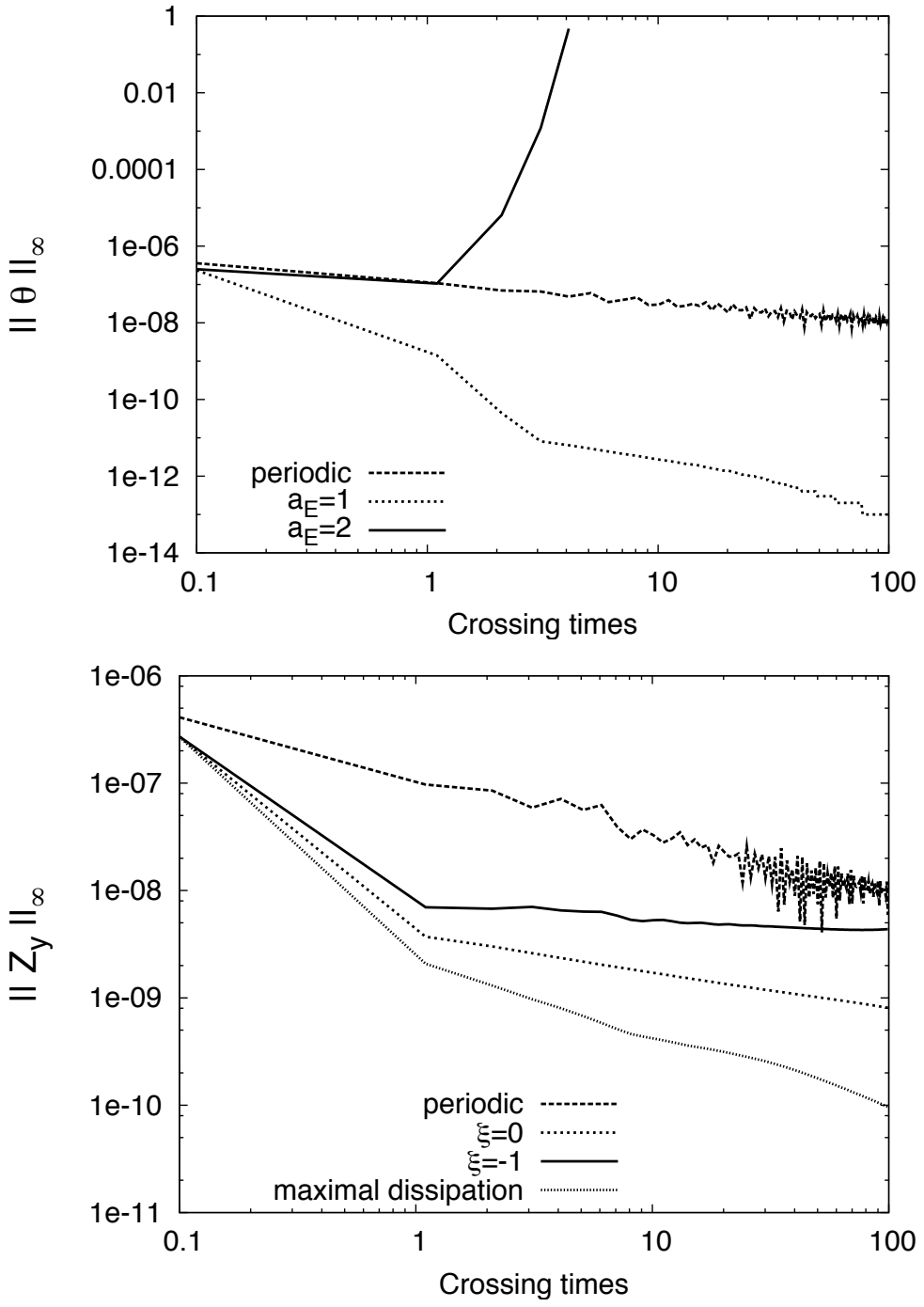


FIGURE 4.3: Robust stability test. Time evolution of the maximum norm of the constraint-violating quantities Θ (up panel), and Z_y (down panel). The proposed boundary conditions are applied here to all faces, including corners and vertices. Some amount of numerical dissipation has been added, so that the periodic boundaries plots (dashed lines) get a visible negative slope. The choice $a_E = 1$ for the energy modes is still clearly stable (upper panel). The choice $\zeta = -1$ for the momentum modes (lower panel) shows a growing mode onset. For comparison, a plot with the maximal dissipation results is also included in the lower panel (bottom line).

Maximal dissipation results are also shown for comparison in the lower panel (bottom line).

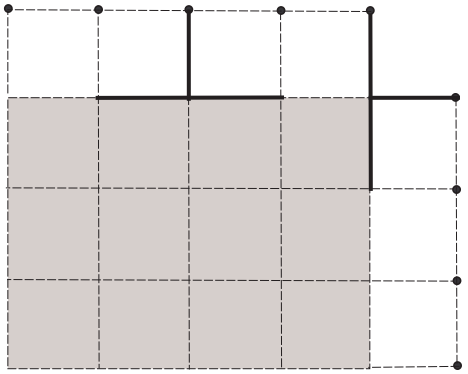


FIGURE 4.4: Stencil for first-order prediction at boundary points (black dots). Interior points belong to the shaded zone. The stencil for two boundary points is represented by thick lines (one space direction has been suppressed for clarity). Note that no corner points are required, as the tangent derivatives are not computed at the boundary, but at the neighbor layer.

We will present here an alternative numerical treatment. At boundary points, tangent derivatives are computed at the next-to-last layer. The corresponding stencil is shown in fig. 4.4. In this way the corner points are not required. This avoids the reported code stability issues, even without adding extra numerical dissipation terms. Note that transverse derivatives are still computed using the standard three-point SBP algorithm, like in the smooth boundaries case. As every space derivative can be considered separately (we are dealing with a first-order system) the SBP property should still follow for our modified scheme. The price for the shift of the transverse derivatives to the next-to-last layer is getting just first-order accuracy at the boundary, but the longitudinal derivatives there were yet only first-order accurate anyway.

This discretization variant allows getting stable results, at least for the value $\zeta = 0$ of the ordering parameter. We plot in fig. 4.5 the maximum norm of the constraint-violation quantities (Θ, Z_i). We can see there that removing the extra numerical dissipation makes the features more transparent. The instability of the $a_E = 2$ choice of the energy coupling parameter, appears now instantly. The downfall rate in the stable case $a_E = 1$, increased as the constraint violations are drained out in all three directions now, can be seen in a more unambiguous way. Concerning the momentum constraint (upper panel), the standard $\zeta = -1$ ordering shows now clearly its unstable behavior, which was masked by the added dissipation in the standard treatment (see fig. 4.3). The centered ordering choice $\zeta = 0$ recovers instead the manifest stable behavior shown in single-face simulations (see fig. 4.2), close to the maximal dissipation case (upper panel, bottom line).

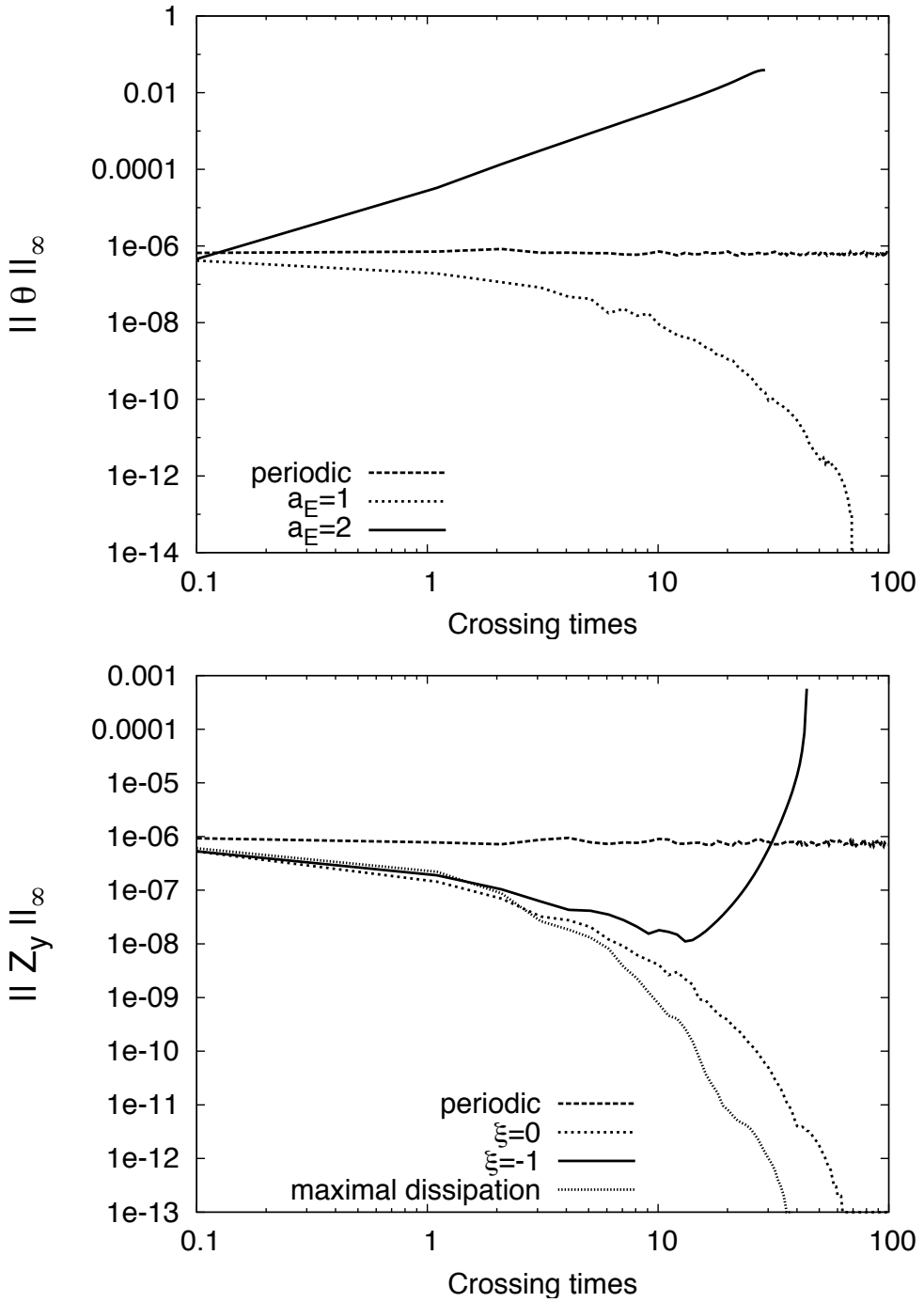


FIGURE 4.5: Robust stability test. Time evolution of the maximum norm of the constraint-violating quantities Θ (upper panel), and Z_y (lower panel). The proposed boundary conditions are applied to all faces. Corner points are avoided in the way shown in fig. 4.4. No extra numerical dissipation has been added, so that the periodic boundaries plots (dashed lines) keep flat. The absence of extra dissipation clarifies the features shown in the previous figure.

Our results show the numerical stability of the proposed boundary conditions in the linear regime for suitable combinations of the coupling and/or ordering parameters. The proposed boundary conditions produce instead a very effective decreasing of (the cumulated effect of) energy and momentum constraint violations which compares with the one obtained by applying maximal dissipation boundary conditions to (the right-hand-side of) the constraint related modes.

4.5 Gowdy waves as a strong field test

Although the results of the preceding are encouraging, let us remark that we were just testing the linear regime around Minkowsky spacetime. This is not enough, as high-frequency instabilities can appear in generic, strong field, situations (see for instance ref. [3]). In order to test the strong field regime, we will consider the Gowdy solution [19], which describes a space-time containing plane polarized gravitational waves. This is one of the test cases that is used in numerical code cross-comparison with periodic boundary conditions [8, 11]. One of the advantages is that it allows for periodic and/or reflecting boundary conditions, which can be applied to the modes which are not in the energy-momentum constraint sector. A first proposal for this selective testing of the constraint-related modes has been presented recently [20].

The Gowdy line element can be written as

$$\mathrm{d}s^2 = t^{-1/2}\, e^{Q/2}\,(-\mathrm{d}t^2 + \mathrm{d}z^2) + t\,(e^{P}\,\mathrm{d}x^2 + e^{-P}\,\mathrm{d}y^2) \tag{4.50}$$

where the quantities Q and P are functions of t and z only and periodic in z, that is [23],

$$P = J_0(2\pi t)\,\cos(2\pi z) \tag{4.51}$$

$$\begin{aligned} Q &= \pi J_0(2\pi)J_1(2\pi) - 2\pi t J_0(2\pi t)J_1(2\pi t)\cos^2(2\pi z) \\ &+ 2\pi^2 t^2[{J_0}^2(2\pi t) + {J_1}^2(2\pi t) - {J_0}^2(2\pi) - {J_1}^2(2\pi)] \end{aligned} \tag{4.52}$$

so that the lapse function $\alpha = t^{-1/4}\, e^{Q/4}$ is constant in space at any time t_0 at which $J_0(2\pi t_0)$ vanishes.

Let us now perform the following time coordinate transformation

$$t \;=\; t_0\, e^{-\tau/\tau_0}, \tag{4.53}$$

so that the expanding line element (4.50) is seen in the new time coordinate τ as collapsing towards the $t = 0$ singularity, which is approached only in the limit $\tau \to \infty$. This 'singularity avoidance' property of the τ coordinate follows from the fact that the

resulting slicing by $\tau = const$ surfaces is harmonic [25]. We will launch our simulations in normal coordinates, starting with a constant lapse $\alpha_0 = 1$ at $\tau = 0$ ($t = t_0$).

The discretization is performed like in the preceding section, but with a space resolution $h = 1/100$. Allowing for the fact that the only non-trivial space dependence in the metric is through $\cos(2\pi z)$, the numerical grid is fitted to the range $0 \leq z \leq 1$. In this way, the exact solution admits either periodic or reflection boundary conditions. We can use these exact boundary conditions as a comparison with the constraint-preserving ones that we are going to test. As the Gowdy metric components depend on just one coordinate, we will apply here the proposed constraint-preserving conditions only to the z faces, keeping periodic boundary conditions along the transverse directions. Also, like in the preceding section, we show the results for the $a_E = a_N = a_T = 1$ coupling parameters combination, although other choices of a_N, $a_T = 1, 2$ lead to similar results.

We start with a simple convergence test. As we know the exact solution (4.50), we can directly compute the relative error of every simulation. Then, only two different resolutions are required for checking convergence. We will take $h = 1/50$ and $h = 1/100$ for our Gowdy wave simulations. We plot in fig. 4.6 the energy-constraint-violation quantity Θ at some early time $t = 10$ (upper panel). We see the expected second-order accuracy at the interior points which are yet causally disconnected with the boundaries. We see just first-order accuracy at the boundary, plus a smooth transition zone. This accuracy reduction at boundaries is inherent to simple SBP algorithms, which require a lower-order discretization at boundary points [16]. One could keep instead the accuracy level of the interior points by using more accurate predictions for boundary values, but at the price of loosing the SBP property. In our test case, however, this issue is not affecting the metric components, even at a much later time $t = 250$, as we can see in the upper panel of fig. 4.6.

We show in fig. 4.7 the results of the $h = 1/100$ resolution simulation for the boundary profiles of Θ and Z_i, indicating the accumulated amount of energy and momentum constraint violations (up and down panels, respectively). We apply in both cases the proposed boundary conditions at the z faces to the constraint-related modes, while keeping exact (reflection) boundary conditions for the other modes. We can see that the constraint-preserving conditions result, in this strong-field test, into an accumulated amount of constraint violations (dotted lines) that is similar or even slightly better than the one produced by the interior points treatment, which can be seen in simulations with (exact) reflection boundaries for all faces (continuous lines). Note that the reflection conditions anchor Z_z to zero at the boundary points, which is always more accurate in this test, although not very useful in more realistic cases.

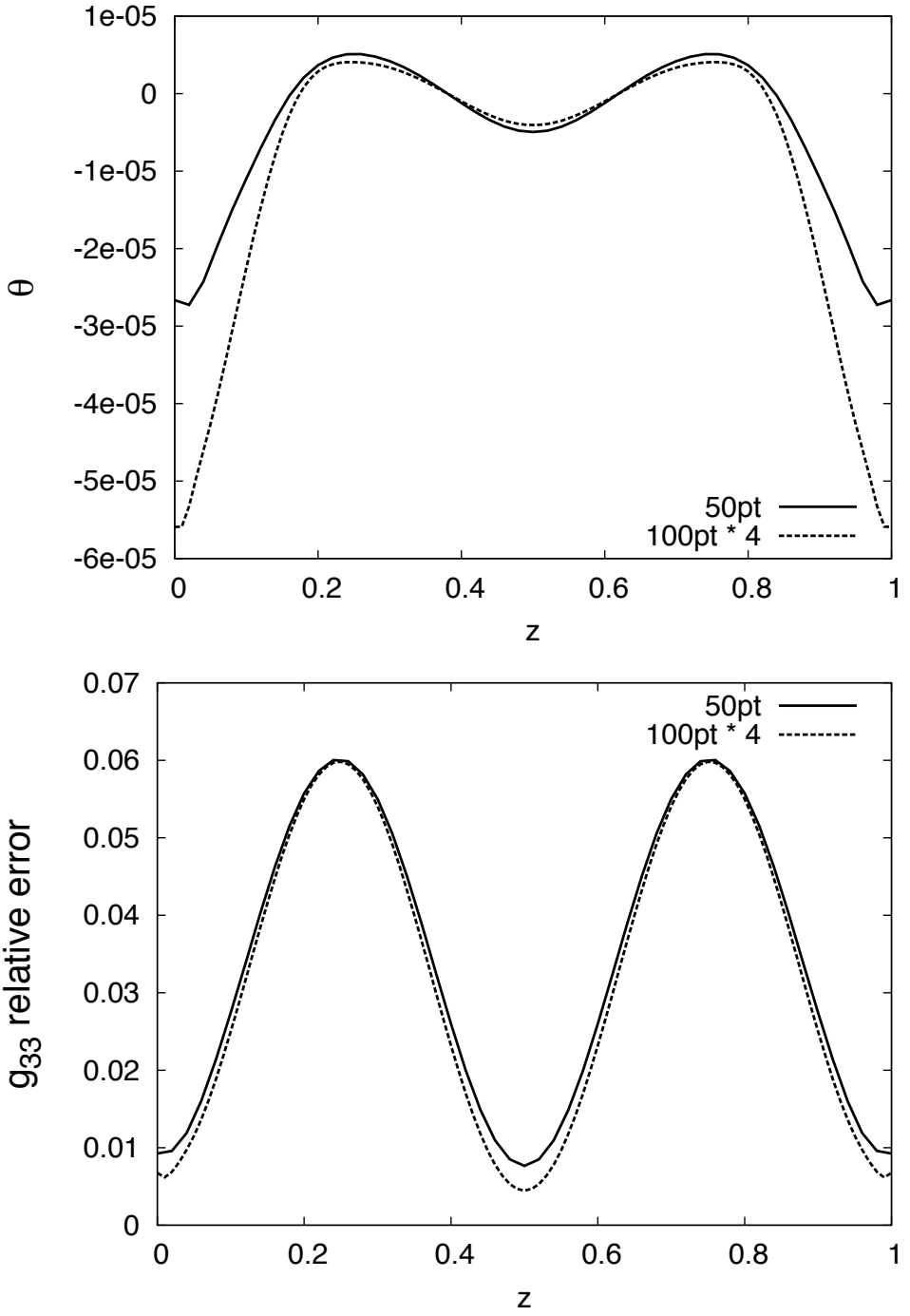


FIGURE 4.6: Convergence test. The constraint-violating quantity Θ is plotted at $t = 10$ for two different resolutions (upper panel). We see second-order convergence in the interior region, decreasing up to first-order at points causally connected to the boundary, as expected from our SBP algorithm. We plot in the lower panel the relative error of the g_{zz} metric component, evolved up to $t = 250$. The boundary-induced accuracy reduction is not even visible yet.

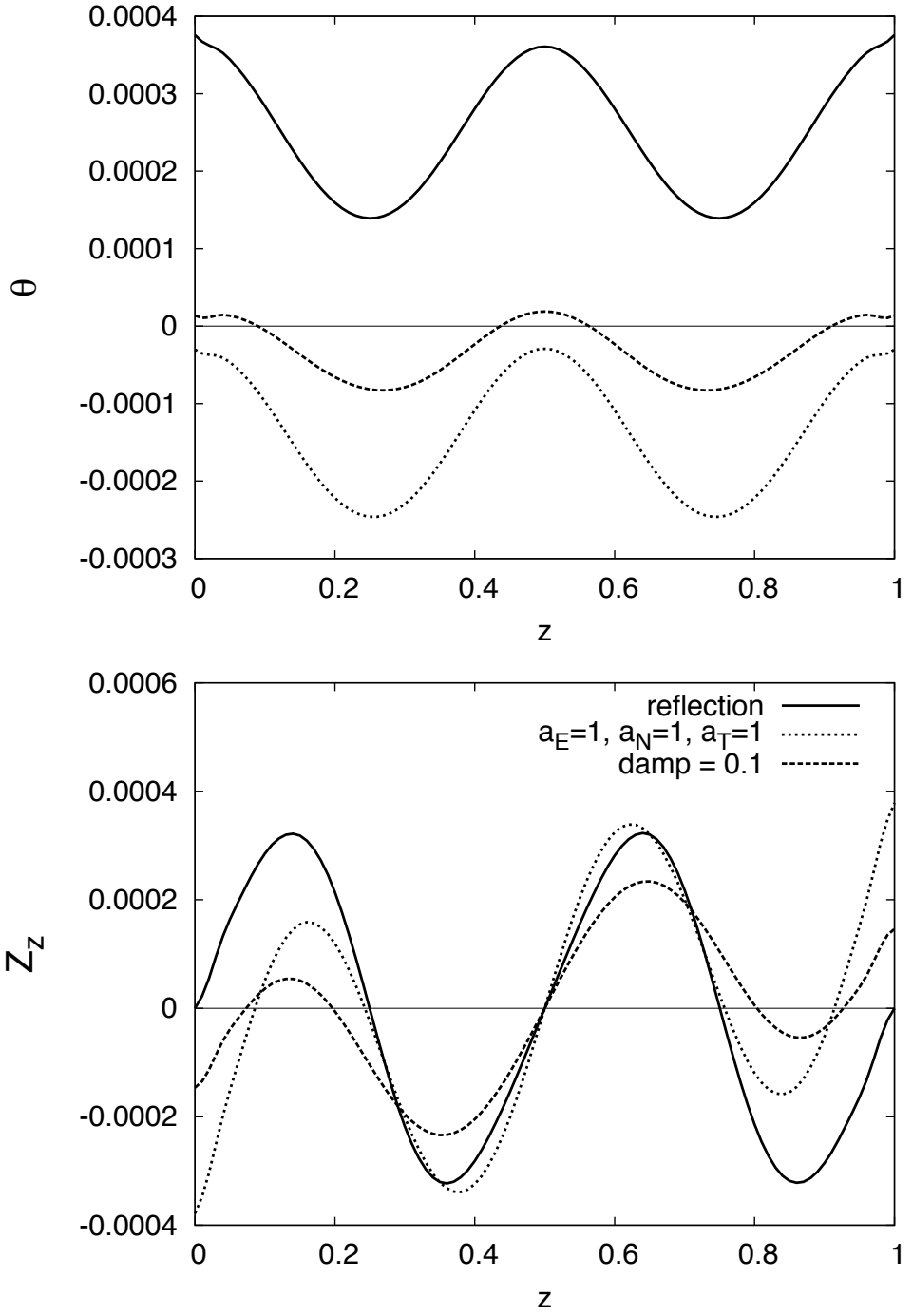


FIGURE 4.7: Gowdy waves test. The Θ and Z_z profiles are plotted as indicators of the accumulated error due to energy-momentum constraint violations. Reflection boundaries results are also plot for comparison (continuous lines). Dotted lines correspond to the proposed boundary conditions, whereas dashed lines correspond to the same conditions with the extra damping terms discussed in this section, with $\eta = 0.1$. All results are shown at t=250

These results confirm that the proposed boundary conditions are indeed constraint-preserving, in the sense that their contribution to energy and momentum constraint violations keeps within the limits of the truncation error of the discretization algorithms, even in this strong field scenario. This good behavior can be further improved by introducing constraint-damping terms in the evolution equations for the boundary quantities (4.36, 4.46) that is

$$\dot{\Theta} \quad + \quad n^k \Theta_k = -\eta \ \Theta \tag{4.54}$$

$$\dot{Z}_i \quad + \quad n^k Z_{ki} = -\eta \ Z_i \, . \tag{4.55}$$

The resulting values can then be used in the replacements (4.39) and (4.49), respectively. We have included the corresponding results in fig. 4.7 (dashed lines). The amount of both energy and constraint violations becomes even lower than the one for the (exact) reflection boundaries simulations even for a small value $\eta = 0.1$ of the damping parameter. The effect is specially visible in the energy constraint case (upper panel).

4.6 Summary

The work presented in this chapter revises and improves the previous results for the Z4 case [5, 15] in many different ways.

On the theoretical side, we have proposed a new symmetrizer, which extends the parametric domain for symmetric hyperbolicity from the single value $\zeta = -1$ to the interval $-1 \leq \zeta \leq 0$. We have identified in the process a new basis for the dynamical field space (4.2-4.5) which allows a clear-cut separation between the constraint-related modes and the remaining ones. Regarding the boundary treatment, we have also generalized the way in which boundary conditions can by used for modifying the incoming modes, by introducing a new parameter a which, at least for the momentum constraint modes, can depart from the standard value $a = 1$ without affecting the stability of the results.

On the numerical side, the use of the new basis definitely improves the stability of the previous Z4 results. In the single face case, where we use periodic boundary conditions along transverse directions, we see that the linear modes previously reported in the robust stability test [5, 15] for the symmetric ordering case ($\zeta = 0$) no longer show up. Moreover, we have devised a simple finite-differences stencil for the prediction step at the boundaries which avoids the corner and vertex points even in cartesian-like grids, providing an interesting alternative to the standard (Olsson) corners treatment.

The proposed boundary conditions have been also tested in a strong field scenario, the Gowdy waves metric [19], so that the effect of non-trivial metric coefficients can be seen

in the simulation results. The convergence test in this non-linear regime provides strong evidence of numerical stability for some suitable parameter combinations. Our simulations actually confirm that the proposed boundary conditions are constraint-preserving: the accumulated amount of energy-momentum constraint violations is similar or even better than the one generated by either periodic or reflection conditions, which are exact in the Gowdy waves case.

Now it remains the question of how these interesting results can be extended to other 3+1 evolution formalisms and/or gauge conditions. Let us remember that all our symmetric hyperbolicity results apply as usual just to the harmonic slicing, not to the '1+log' class of slicing conditions which are customary in BSSN black-hole simulations. There is no problem, however, in extending the proposed boundary conditions to this case: in our new basis the gauge sector is clearly separated from the constraint-related one, so that one can keep using the replacements (4.39, 4.49) even in this non-harmonic case. The shift, however, introduces new couplings and would require a detailed case-by-case investigation: even the strong hyperbolicity of the system can depend on the specific choice of shift condition.

Concerning the extension from the Z4 to the BSSN formalism, the momentum constraint treatment can be derived from the simple condition [15]

$$\widetilde{\Gamma}_i = -\widetilde{\gamma}_{ik}\,\widetilde{\gamma}^{jk}{}_{,j} + 2\,Z_i \tag{4.56}$$

which relates the additional BSSN quantity $\widetilde{\Gamma}_i$ with the space vector Z_i. The replacement (4.49) can then be used for getting a suitable boundary condition in this context. The case of the energy constraint is more challenging, as the BSSN formalism does not contain any supplementary quantity analogous to Θ. One could follow, however, the line recently proposed in [13]: a slight modification of the original BSSN equations allows to include the new quantity Θ, so that the correspondence with the Z4 formalism is complete. The replacement (4.39) can then be used directly in such context.

A major challenge is posed by the fact that most BSSN implementations are of second order in space. This has some advantages in this context, as the ordering constraints do not show up and this removes the main source of ambiguities in the constraint-violations evolution system. As a result, the boundary conditions (4.36, 4.46) become simply advection equations so that we can expect a more effective constraint-violation 'draining' rate. The problem, however, is that second-order implementations do not have the algebraic characteristic decomposition which is crucial in the first-order ones. The boundaries treatment takes quite different approaches in second-order formalisms, although the evolution equations for the constraint-related quantities Θ , Z_i are still of

first order in the Z4 case, even at the continuum level, and this suggests that the results presented here can be still helpful in this case.

References

[1] J. M. Stewart, Class. Quantum Grav. **15** 2865 (1998).

[2] G. Calabrese et al, Commun. Math. Phys. **240** 377 (2003).

[3] O. Sarbach and M. Tiglio, J. Hyperbol. Diff. Equat. **2**, 839 (2005). ArXiv: gr-qc/0412115.

[4] C. Gundlach and JM. Martín–García, Phys. Rev. D**70** 044032 (2004).

[5] C. Bona, T. Ledvinka, C. Palenzuela, M. Žáček, Class. Quantum Grav. **22**, 2615 (2005).

[6] F. Pretorius, Phys. Rev. Lett. **95**, 121101 (2005).

[7] M. Babiuc, H–O. Kreiss and J. Winicour, Phys. Rev. D**75** 044002 (2007).

[8] H–O. Kreiss, O. Reula, O. Sarbach and J. Winicour, Class. Quantum Grav.**24** 5973 (2007).

[9] J. Winicour, Gen. Rel. Grav. **41**, 1909–1926 (2009).

[10] O. Rinne, L. Buchman, M. Scheel and H. Pfeiffer, Class. Quantum Grav. **26** 075009 (2009).

[11] M. Ruiz, D. Hilditch and S. Bernuzzi, Phys. Rev. D**83** 024025 (2011)

[12] D. Núñez and O. Sarbach, Phys. Rev. D**81** 044011 (2010).

[13] S. Bernuzzi and D. Hilditch, Phys. Rev. D **81**, 084003 (2010).

[14] C. Gundlach, G. Calabrese, I. Hinder and J.M. Martín-García, Class. Quantum Grav. **22**, 3767 (2005).

[15] C. Bona and C. Palenzuela, *Elements of Numerical Relativity*, Lect. Notes Phys. **673** (Springer, Berlin–Heidelberg–New York 2005).

[16] P. Olsson, Mathematics of Computation **64**, 1035 (1995).

[17] C. Bona, C. Bona-Casas and J. Terradas, J. Comp. Physics **228**, 2266 (2009).

[18] D. Alic, C. Bona and C. Bona–Casas, Phys. Rev. D **79**, 044026 (2009).

[19] R. H. Gowdy, Phys. Rev. D **27**, 826 (1971).

[20] C. Bona and C. Bona–Casas, J. Physics: Conference Series. 229 012022, (2010). ArXiv: 0911.1208

[21] C. Bona, T. Ledvinka, C. Palenzuela and M. Žáček, Phys. Rev. D **69**, 064036 (2004).

[9] C. Bona, T. Ledvinka, C. Palenzuela, M. Žáček, Phys. Rev. D **67**, 104005 (2003).

[23] M. Alcubierre et al, Class. Quantum Grav. **21**, 589 (2004).

[24] O. A. Liskovets, J. Differential Equations **I**, 1308-1323 (1965).

[25] C. Bona and J. Massó, Phys. Rev. **D38** 2419 (1988).

Chapter 5

Further Developments

We might also find interesting things if we take a completely different approach. There is a growing interest in incorporating the new hyperbolic formulations into the Lagrangian/Hamiltonian framework. An example is the usage of the 'densitized lapse' [1] as a canonical variable, leading to a modification in the standard form of the canonical evolution equations [2]. Reciprocally, there are very recent attempts of modifying the ADM action [3] in order to incorporate coordinate conditions of the type used in numerical relativity [4, 5], with a view on using symplectic integrators for the time evolution, which could ensure constraint preservation in numerical simulations [6]. On a different context, a well posed evolution formalism developed from a Lagrangian formulation could be a good starting point for Quantum Gravity applications.

In this chapter we derive the Z4 formalism from an action principle by introducing a Lagrangian density which generalizes the Hilbert action for Einstein's equations. This is a crucial step towards the Lagrangian formulation of other numerical-relativity formalisms. We actually consider here also the BSSN case, following the symmetry-breaking mechanism described in refs. [7, 8]. At this point we must say there is a lot of literature discussing the mutual equivalence between the Hilbert action, which is in the 4D framework, and the ADM one, which is in the 3+1 framework. The point is that ADM excludes lapse and shift from the dynamical quantities, whereas Hilbert takes all g_{ab} components. All calculations that are action-related in this chapter will be in 4D.

In the last section, we provide a conformal decomposition of the Z4 system, mimicking the BSSN one. We see that this allows us to perform simulations with 'puncture data', at least for the single BH case.

5.1 Generalizing the Einstein-Hilbert action principle

Let us consider the generic action

$$S = \int d^4x \; \mathcal{L} \tag{5.1}$$

with a Lagrangian density which generalizes the Einstein-Hilbert one by including an extra four-vector Z_μ, namely

$$\mathcal{L} = \sqrt{g} \; g^{\mu\nu} \left[R_{\mu\nu} + 2 \nabla_\mu Z_\nu \right] \tag{5.2}$$

(we restrict ourselves to the vacuum case), with the Ricci tensor written in terms of the connection coefficients

$$R_{\mu\nu} = \partial_\rho \Gamma^\rho_{\mu\nu} - \partial_{(\mu} \Gamma^\rho_{\nu)\rho} + \Gamma^\rho_{\rho\sigma} \, \Gamma^\sigma_{\mu\nu} - \Gamma^\rho_{\sigma\mu} \, \Gamma^\sigma_{\rho\nu} \; , \tag{5.3}$$

(round brackets denote symmetrization). The standard definition is without symmetrization, but only the symmetric part contributes to the Lagrangian (assuming $g_{\mu\nu}$ symmetric).

Now let us follow the well-known Palatini approach, by considering independent variations of the metric density $h^{\mu\nu} = \sqrt{g} \; g^{\mu\nu}$, the connection coefficients $\Gamma^\rho_{\mu\nu}$ and the vector Z_μ. This way we are taking into account that connection coefficients do not have to necessarily be metric connection coefficients. From the $h^{\mu\nu}$ variations we get directly the Z4 field equations [9] (strictly speaking, they will be not Z4 equations until connection coefficients will be imposed to be metric connection coefficients)

$$R_{\mu\nu} + \nabla_\mu Z_\nu + \nabla_\nu Z_\mu = 0 \, , \tag{5.4}$$

which are currently used in many numerical-relativity developments as one can see in this thesis and in [8]. We will use the abbreviation

$$\Omega^\rho{}_{\mu\nu} = \Gamma^\rho_{\mu\nu} - \bar{\Gamma}^\rho_{\mu\nu} \tag{5.5}$$

for the difference between the generic connection $\Gamma^\rho_{\mu\nu}$ and the metric one: the Christoffel symbols $\hat{\Gamma}^\rho_{\mu\nu}$. Variation with respect to Z_μ and $\Gamma^\rho_{\mu\nu}$ yield the following equations

$$0 = \frac{1}{\sqrt{g}}\frac{\delta\mathcal{L}}{\delta Z_\mu} = -2\Omega^{\mu\sigma}{}_{\sigma} \tag{5.6}$$

$$0 = \frac{1}{\sqrt{g}}\frac{\delta\mathcal{L}}{\delta\Gamma^{\rho}_{\mu\nu}} = \Omega^{\rho}{}_{\rho\sigma}g^{\mu\nu} - 2\Omega^{(\mu\nu)}{}_{\sigma} + \delta^{(\mu}_{\sigma}\Omega^{\nu)\rho}{}_{\rho} - 2Z_\sigma g^{\mu\nu} \tag{5.7}$$

We can now solve for $\Omega^{\rho}_{\mu\nu}$. Let us take the trace over the indices μ and ν in (5.7) to yield

$$\Omega^{\rho}{}_{\rho\mu} = \frac{10}{3}Z_\mu \tag{5.8}$$

Putting the results (5.7-5.8) together gives

$$\Omega_{\mu\nu\sigma} + \Omega_{\nu\mu\sigma} = \frac{4}{3}(Z_\sigma g_{\mu\nu} + Z_{(\mu}g_{\nu)\sigma}) \tag{5.9}$$

Now we write down two more copies of this equation with index replacements $\mu \to \nu$, $\nu \to \sigma$, $\sigma \to \mu$ in the first copy and $\mu \to \sigma, \nu \to \mu, \sigma \to \nu$ in the second. We add the second copy to (5.9) and then we substract the first copy. With this we obtain

$$\Omega^{\sigma}{}_{\mu\nu} = \frac{4}{3}\delta^{\sigma}_{(\mu}Z_{\nu)} \tag{5.10}$$

as a solution for (5.7). With this result we see that equations (5.6, 5.7) put together have the solution

$$\Omega^{\rho}{}_{\mu\nu} = 0 \leftrightarrow \nabla_\rho\, g^{\mu\nu} = 0\,, \tag{5.11}$$

which fixes the connection coefficients in terms of the metric, and the vector condition

$$Z_\mu = 0. \tag{5.12}$$

Let us note here the different role of the conditions (5.11) and (5.12). As there are much more independent connection coefficients than evolution equations in (5.4), we will consider condition (5.11) as a constraint enforcing the metric connection 'a posteriori', that is after the variation process. In this way, we will ensure that equations (5.4) are identical to the original Z4 equations, rather than some affine generalization. For this reason, we will assume a metric connection everywhere in what follows.

The case of condition (5.12) is different, as the Z4 equations (5.4) actually provide evolution equations for every component of Z_μ. Then, (5.12) is a standard primary constraint and we have a choice among different strategies for dealing with it. If we enforce (5.12) into the Z4 field equations (5.4), we get nothing but Einstein's equations. This is not surprising because our Lagrangian obviously reduces to the Einstein-Hilbert one when Z_μ vanishes. The problem is that the plain Einstein field equations do not lead directly to a well-posed initial data problem. This is why the original harmonic formulation [10–12] was used instead in the context of the Cauchy problem [13]. For the same reason, other formulations (BSSN [14, 15], generalized harmonic [16–18], Z4 [7, 9]) are currently considered in numerical relativity.

5.2 Recovering the Z4 formulation

We can alternatively follow a different strategy. Instead of enforcing (5.12), we can deal with this condition as an algebraic restriction to be imposed just on the initial data, that is

$$Z_\mu \,|_{t=0} = 0 \tag{5.13}$$

In this way, we can keep the Z4 field equations system, which is known to be strongly hyperbolic when supplemented by gauge conditions, like '1+log' or 'freezing shift', suitable for numerical evolution [7, 8]. The consistency of this 'relaxed' approach requires that the constraint (5.12) should be actually preserved by the Z4 field equations (5.4). In this way, the solutions obtained from initial data verifying (5.12) will actually minimize the proposed action (5.1).

Allowing for the conservation of the Einstein tensor, which is granted after the metric connection enforcement, we derive from (5.4) the second-order equation, linear-homogeneous in Z

$$\nabla_\nu \left[\nabla^\mu Z^\nu + \nabla^\nu Z^\mu - (\nabla_\rho Z^\rho)\, g^{\mu\nu} \right] = 0 \,. \tag{5.14}$$

It follows that the necessary and sufficient condition for the preservation of the constraint (5.12) is to impose also its first time-derivative conditions in the initial data, that is

$$(\partial_0 \, Z_\mu) \,|_{t=0} = 0 \,. \tag{5.15}$$

Note that, allowing for (5.13) and the Z4 field equations, the secondary constraints (5.15) amount to the standard energy and momentum constraints, which are then to be imposed on the initial data in addition to (5.13).

This 'relaxed' treatment of the constraints (5.12) may look unnatural. But it is just the reflection of a common practice numerical relativity ('free evolution' approach), where four of the ten field equations (the energy-momentum constraints) are not enforced during the evolution, being imposed just in the initial data instead. The introduction of the extra four-vector in the Z4 formalism actually provides a simpler implementation of the same idea.

5.3 Recovering the Z3-BSSN formulation

Note that in all our developments we have preserved general covariance. Our action integral (5.1) is a true scalar and, in spite of other alternatives, we have avoided the addition of total divergences which could have simplified our developments to some extent, at the price of adding boundary terms. This means that we keep at this point the full coordinate-gauge freedom.

However, as it is well known, the BSSN formulation is not general covariant. It contains just three additional 'contracted-gamma' quantities, associated to the momentum constraint. But the 3+1 splitting between energy and momentum requires a specific choice for the time coordinate, which breaks general covariance. In refs. [7, 8] a symmetry breaking mechanism is proposed, which allows to recover BSSN from the 'Z3 system' [19], which is obtained in turn from the Z4 one by enforcing the energy constraint in the form

$$Z^0 = 0\,. \tag{5.16}$$

We can proceed now like in the preceding section, by the following steps:

- Enforcing both the metric connection condition (5.11) and (5.16) in the Z4 field equations.
- Splitting the ten resulting equations into the (energy-related) secondary constraint

$$\mathcal{E} \equiv \partial_t Z^0 = 0 \tag{5.17}$$

  plus nine evolution equations for the six space metric components and the space vector Z_i.

Note that this splitting is not unique, as any multiple of $\mathcal{E}$ (times the metric) can be added to the evolution equations, resulting into a modified evolution system, with a different principal part. In this way we obtain the one-parameter family of (generalized)

Z3 evolution systems [7, 8]. For a specific value of the parameter, one can recover a flavor of the BSSN formalism (Z3-BSSN) by means of a conformal decomposition of the spatial tensors and the redefinition of space vector Z_i in terms of the equivalent 'contracted-Gamma' quantities, namely

$$\tilde{\Gamma}_i = -\tilde{\gamma}_{ik}\ \tilde{\gamma}^{jk}_{,j} + 2\ Z_i \tag{5.18}$$

where $\tilde{\gamma}_{ij}$ is the conformal space metric. The space vector Z_i can be interpreted in this context as the difference between the (contracted) conformal metric connection and the BSSN contracted-Gamma quantities. This provides actually time a three-covariant reformulation of the original BSSN formalism.

5.4 Generalized Harmonic systems

There is still another possibility, which allows a more direct specification of a coordinate gauge at the price of breaking the covariance of the evolution equations. which are currently used in many numerical-relativity developments [8]. We can enforce in the Z4 equations (5.4) the following assignment for Z_μ

$$Z^\mu = -\frac{1}{2}\ \Gamma^\mu_{\rho\sigma}\, g^{\rho\sigma} \equiv -\frac{1}{2}\ \Gamma^\mu\,. \tag{5.19}$$

The vanishing of Z_μ will amount in this way to the 'harmonic coordinates' condition, which can be considered then as a constraint to be imposed just in the initial data, that is

$$(\Gamma^\mu{}_{\rho\sigma}\, g^{\rho\sigma})\,|_{t=0} = 0 \tag{5.20}$$

(note that the extra field Z_μ has disappeared in the process). The resulting field equations

$$R_{\mu\nu} - \ \partial_{(\mu}\Gamma_{\nu)} + \Gamma^\rho_{\mu\nu}\,\Gamma_\rho = 0 \tag{5.21}$$

lead, after imposing the metric connection condition (5.11), to the manifestly hyperbolic second-order system

$$g^{\rho\sigma}\partial^2_{\rho\sigma}\ g_{\mu\nu} = 2\, g^{\rho\sigma} g^{\alpha\beta}\,[\partial_\alpha g_{\rho\mu}\ \partial_\beta g_{\sigma\nu} - \Gamma_{\mu\rho\alpha}\ \Gamma_{\nu\sigma\beta}]\,. \tag{5.22}$$

This corresponds to the classical harmonic formulation of General Relativity [10–12], which is known to have a well-posed Cauchy problem [13].

We have derived in this way the harmonic formalism through the non-covariant prescription (5.19). The harmonic constraint (5.20) is automatically preserved by the resulting

(harmonic) evolution system, provided that we also enforce the energy-momentum constraints on the initial data. This can be seen in a transparent way by replacing directly (5.19) into the covariant constraint-evolution equation (5.14) and then into the resulting conditions (5.15).

The prescription (5.19) can be generalized in order to enforce other coordinate gauges that are also currently used in numerical relativity. The simpler formulations [16, 18] correspond to the replacement

$$Z^\mu = -\frac{1}{2}\,\left(\,\Gamma^\mu + H^\mu\,\right), \tag{5.23}$$

where the 'gauge sources' H^μ are explicit functions of the metric and/or the spacetime coordinates. If we follow the same process the resulting field equations are

$$R_{\mu\nu} - \,\partial_{(\mu}\Gamma_{\nu)} - \,\partial_{(\mu}H_{\nu)} + \Gamma^\rho_{\mu\nu}\,\Gamma_\rho + \Gamma^\rho_{\mu\nu}\,H_\rho = 0 \tag{5.24}$$

And, again, imposing the metric connection condition (5.11), we obtain

$$g^{\rho\sigma}\partial^2_{\rho\sigma}\;g_{\mu\nu} + \,\partial_{(\mu}H_{\nu)} - \Gamma^\rho_{\mu\nu}\,H_\rho = 2\,g^{\rho\sigma}g^{\alpha\beta}\,[\partial_\alpha g_{\rho\mu}\;\partial_\beta g_{\sigma\nu} - \Gamma_{\mu\rho\alpha}\;\Gamma_{\nu\sigma\beta}]\,. \tag{5.25}$$

More general choices of H^μ, like that of ref. [20], would require a more elaborate treatment.

The same mechanism can be applied to coordinate conditions derived in the 3+1 framework, where the spacetime line element is decomposed as

$$ds^2 = -\alpha^2\;dt^2 + \gamma_{ij}\;(dx^i + \beta^i\;dt)\;(dx^j + \beta^j\;dt)\,. \tag{5.26}$$

The spacetime slicing is given by the choice of the time coordinate. In this context, the harmonic slicing condition can be generalized to [7]

$$(\partial_t - \beta^k\partial_k)\;\alpha = -f\,\alpha^2\;trK\,, \tag{5.27}$$

where $K_{ij} = -\alpha\,\Gamma^0_{ij}$ stands for the extrinsic curvature of the time slices. The case $f = 1$ corresponds to the harmonic time-coordinate choice, whereas the choice $f = 2/\alpha$ corresponds to the popular '1+log' time slicing.

In order to get the replacement, of the form (5.23), which connects this condition with our formulation, we must rewrite (5.27) in a four-dimensional form with the help of the

$\Gamma_{nnn} = 1/\alpha^2\,(\partial_t - \beta^r\partial_r)\,\alpha$	$\Gamma_{nnk} = -\partial_k ln\,\alpha$
$\Gamma_{knn} = 1/\alpha^2\,\gamma_{kj}(\partial_t - \beta^r\partial_r)\beta^j + \partial_k ln\,\alpha$	$\Gamma_{nij} = -K_{ij}$
$\Gamma_{ijn} = K_{ij} - 1/\alpha\,\gamma_{ik}\,\partial_j\,\beta^k$	$\Gamma_{kij} = {}^{(3)}\Gamma_{kij}$

TABLE 5.1: The 3+1 decomposition of the four-dimensional connection coefficients. The index n is a shorthand for the contraction with the unit normal n_μ.

unit normal n_μ to the constant time hypersurfaces, that is

$$n_\mu = \alpha\;\delta^0_\mu \qquad n^\mu = (-\delta^\mu_0 + \delta^\mu_i\beta^i)/\alpha\,. \tag{5.28}$$

We can now decompose the four-dimensional Christoffel symbols in terms of the standard 3+1 quantities (see Table 5.1). This provides a convenient way of translating 3+1 conditions like (5.27) in terms of four-dimensional objects.

We can obtain in this way, after an straightforward calculation, the gauge sources corresponding to the class of slicing conditions (5.27), namely

$$H^0 = (1 - 1/f)\;\Gamma^0{}_{\rho\sigma}n^\rho n^\sigma\,. \tag{5.29}$$

We will use now (5.23) for replacing the quantity Z^0 in the Z4 equations. Its evolution equation gets transformed in this way into a second order evolution equation for the lapse function α, which governs the spacetime slicing. As the first-order slicing condition (5.27) has been translated into a specification of Z^0, and allowing for (5.14), (5.27) will become a first integral of the second order evolution system: we can impose it just in the initial data together with the energy-momentum constraints. This approach is new in 3+1 formalisms, but a common practice in the harmonic-like ones.

The same technique can be used for 'gamma-driver' shift prescriptions. A first-order reduction of the original 'gamma-freezing' condition [21] is given by [22]

$$(\partial_t - \beta^k\partial_k)\;\beta^i = \mu\;\tilde{\Gamma}^i - \eta\;\beta^i\,, \tag{5.30}$$

where $\tilde{\Gamma}^i$ stands here for the contraction of the three-dimensional conformal connection, that is

$$\tilde{\Gamma}^i \equiv \gamma^{ij}\,\gamma^{rs}\,(\,\Gamma_{j\,rs} + \frac{1}{3}\;\Gamma_{rsj}\,)\,. \tag{5.31}$$

The corresponding 'gamma-driver' gauge sources are given by

$$\begin{aligned} H_i &= (1 - \frac{\alpha^2}{\mu})\,\Gamma_{i\rho\sigma}n^\rho n^\sigma + \frac{1}{3}\;\Gamma_{\rho\sigma i}\,g^{\rho\sigma} \\ &+ (\frac{1}{3} - \frac{\alpha^2}{\mu})\,\Gamma_{\rho\sigma i}n^\rho n^\sigma - \eta/\mu\;g_{0i}\,. \end{aligned} \tag{5.32}$$

We can use again (5.23), this time for replacing the space vector Z_i in the Z4 equations. Its evolution equation get transformed in this way into a second order evolution equation for the shift components β^i, which determine the time lines. Again, the first-order gamma-driver condition (5.30) becomes a first integral of the resulting (second order) shift evolution equation. At the same time, one gets rid of the additional variables Z_i (as we did for Z^0 with the analogous replacement, leading to the lapse evolution equation).

5.5 Conformal Z4

By the end of the past century, some hyperbolic extensions of Einstein's equations were developed with a view on numerical relativity applications [1, 23–25]. This emergent field is now more mature: there are two main formalisms currently used in numerical simulations. One is BSSN [14, 15], working at the 3+1 level, and the other is the class of generalized harmonic formalisms [16–18], working at the four-dimensional level. A unifying framework is provided by the Z4 formalism [9], which is precisely the one we have been using in the last two chapters and it allows to recover the generalized harmonic one by relating the additional vector field Z_μ with the harmonic 'gauge sources' [16]. It also allows to recover (a specific version of) BSSN by a symmetry-breaking process in the transition from the four-dimensional to the three-dimensional formulations [7, 8].

The development of these formalisms has been very useful for the field, which achieved a major breakthrough by successfully simulating binary black hole scenarios after many years of research. Many research groups have therefore since then focused their efforts in exploring the possibilities of their working codes: different mass ratios, spinning and non-spinning black holes... These results are now being gathered (see for example the recent NRAR, Numerical Relativity-Analytical Relativity collaboration) in order to obtain different gravitational waves signal patterns for the data analysis community to use and some standards are being set.

But after this fruitful period of carrying out simulations and exploring the limits of the available tools for numerical codes (mesh refinement and parameter tweaking in general) numerical relativity is facing new walls which prevent the field from evolving. Extreme mass ratio simulations, precision of the results, higher spins, higher number of orbits... they are all limited by computational power and efficiency. In this thesis we have presented different tools that might be useful to overcome these limitations, like adopting a new numerical method (FDOC), constraint preserving boundary conditions that might allow to put the boundaries of the domain closer and we have also shown the numerical robustness of the Z4 formalism (besides its nice mathematical properties,

that were already known), which has now become a good candidate to perform binary black hole simulations.

In fact, after Z4 has shown that is able to successfully perform single black hole simulations with scalar field initial data (see Chapter 3 and [26]), there is a renewed interest in the formalism and some work has been done in order to get Z4 to perform single black holes in spherical symmetry with puncture initial data and also deal with matter spacetimes [27]. In that paper, a conformal decomposition of the second order Z4 system was presented, but many source terms (non principal part terms) were dropped. Therefore the resulting system, though it is called Z4c, lacks 4-covariance. The authors justify it with the intention of obtaining a system very similar to BSSN (and therefore very easy to implement in the existing BSSN codes) and they claim to obtain a much better keeping of the Hamiltonian constraint with matter spacetimes when they compare the Z4c results with the BSSN results.

With all these evidence put together, there are many reasons to think that a 4-covariant conformal decomposition of the Z4 system will be able to perform 3D black holes with puncture initial data and a freezing shift condition (therefore extending the polyvalence of the system regarding the gauge choice and initial data) and will also be able to simulate binary black hole systems.

Let us start from the second order Z4 system (3.3-3.6). For the sake of simplicity, we will exclude the shift terms from the forthcoming calculations. If we use (3.3) and that:

$$\begin{aligned} \gamma^{is}\ \partial_t\ (\gamma^{rj}\gamma_{ij}) &= \gamma^{is}\ \partial_t\ (\delta^r_i) = 0 \\ &= \gamma^{is}\ \gamma^{rj}\ \partial_t\ (\gamma_{ij}) + \gamma^{is}\ \gamma_{ij}\ \partial_t\ (\gamma^{rj}) \\ &= \gamma^{is}\ \gamma^{rj}\ (-2\ \alpha\ K_{ij}) + \delta^s_j\ \partial_t\ (\gamma^{rj}) \end{aligned} \tag{5.33}$$

We can obtain the following:

$$\partial_t\ \gamma^{ij} = 2\ \alpha\ K^{ij} \tag{5.34}$$

So with (5.34) and (3.4) we can obtain an evolution equation for the trace of the extrinsic curvature:

$$\begin{aligned} \partial_t\ K &= \partial_t\ (\gamma^{ij} K_{ij}) = \gamma^{ij}\ \partial_t\ K_{ij} + K_{ij}\ \partial_t\ \gamma^{ij} = \\ &= -\nabla^i \alpha_i + \alpha\ [R + 2\ \nabla^i Z_i + K^2 - 2\ \theta K] \end{aligned} \tag{5.35}$$

Where $K = K^i_i$ and $R = R^i_i$. Now we need to obtain an evolution equation for the trace-free part of the extrinsic curvature, which is defined as follows

$$A_{ij} = K_{ij} - \frac{1}{3}\gamma_{ij}\ K \tag{5.36}$$

And if we take a time derivative in (5.36) and we use (3.3, 3.4) and (5.35):

$$\begin{aligned} \partial_t\ A_{ij} &= \partial_t\ K_{ij} - \frac{1}{3}\gamma_{ij}\ \partial_t\ K - \frac{1}{3}K\ \partial_t\ \gamma_{ij} = \\ &= [-\nabla_i \alpha_j + \alpha\ (R_{ij} + \nabla_i Z_j + \nabla_j Z_i)]^{TF} \\ &\quad +\alpha(K - 2\theta)(A_{ij} + \frac{1}{3}\gamma_{ij}K) - 2\alpha(A_{il} + \frac{1}{3}\gamma_{il}K)(A^l_j + \frac{1}{3}\delta^j_l K) \\ &\quad +\frac{2}{3}K\alpha(A_{ij} + \frac{1}{3}\gamma_{ij}K) = \\ &= [-\nabla_i \alpha_j + \alpha(R_{ij} + \nabla_i Z_j + \nabla_j Z_i)]^{TF} + \alpha(\frac{1}{3}K A_{ij} - 2\theta A_{ij} - 2A_{il}A^l_j) \end{aligned} \tag{5.37}$$

Where TF denotes that only the trace-free part of the terms is involved in the calculations.

To proceed, we need a splitting for the spatial metric γ compatible with the splitting of K_{ij} into K and A_{ij}. In the BSSN formulation, the desired splitting is achieved by introducing the conformal factor and the conformal metric

$$\phi = \frac{1}{12} ln\gamma \qquad \tilde{\gamma}_{ij} = e^{-4\phi}\gamma_{ij} \tag{5.38}$$

Where γ is the determinant of the metric. However, a variable of the form

$$\chi = \gamma^{-1/\kappa} \qquad \tilde{\gamma}_{ij} = \gamma^{-1/3}\gamma_{ij} = \chi^{\kappa/3}\gamma_{ij} \tag{5.39}$$

Has been suggested [28, 29]. In [28], it is noted that certain singular terms in the evolution equations for Bowen-York initial data can be corrected taking $\kappa = 3$. Alternatively, [29] notes that $\kappa = 6$ has the additional benefit of ensuring that γ remains

positive, a property which needs to be explicitly enforced with $\kappa = 3$. We can follow the same approach with Z4 and if we make use of the Leibnitz formula to differentiate the determinant of a matrix

$$\partial\ \gamma = \gamma\ \gamma^{lm}\ \partial\ \gamma_{ml} \tag{5.40}$$

Using (5.40) and (3.3) we find that the evolution equation for the conformal factor is

$$\partial_t\ \chi = \partial_t\ (\gamma^{-1/\kappa}) = -\frac{1}{\kappa}\gamma^{-1/\kappa}\gamma^{lm}\partial_t\ \gamma_{ml} = \frac{2}{\kappa}\alpha\chi K \tag{5.41}$$

Now using (5.41, 3.3, 3.4 and 5.36) we can obtain the evolution equation for the conformal metric

$$\begin{aligned}\partial_t\ \tilde{\gamma}_{ij} &= \partial_t\ (\chi^{\kappa/3}\gamma_{ij}) = \chi^{\kappa/3}\partial_t\ \gamma_{ij} + \frac{\kappa}{3}\gamma_{ij}\chi^{\kappa/3-1}\partial_t\ (\chi) = \\ &= \chi^{\kappa/3}(-2\alpha K_{ij} + \frac{2}{3}\alpha\gamma_{ij}K) = -2\alpha\tilde{A}_{ij}\end{aligned} \tag{5.42}$$

Where $\tilde{A}_{ij} = \chi^{\kappa/3}A_{ij}$ is the conformal analog of the variable A_{ij}. The evolution equation for $\tilde{A}_{ij}$ is then:

$$\begin{aligned}\partial_t\ \tilde{A}_{ij} &= \partial_t\ (\chi^{\kappa/3}A_{ij}) = \chi^{\kappa/3}[\partial_t\ A_{ij} + \frac{2}{3}\alpha K A_{ij}] = \\ &= \chi^{\kappa/3}[-\nabla_i\alpha_j + \alpha(R_{ij} + \nabla_i Z_j + \nabla_j Z_i)]^{TF} \\ &\quad +\alpha\tilde{A}_{ij}(K - 2\theta) - 2\alpha\tilde{A}_{il}\tilde{A}^l_j\end{aligned} \tag{5.43}$$

The gauge quantities, α and β^i, will be evolved using the prescriptions that have been commonly applied to BSSN black hole, and particularly puncture, evolutions. For the lapse, we will evolve according to the "1+log" condition [30],

$$\partial_t\ \alpha = -2\alpha(K - m\theta) + \beta^i\partial_i\ \alpha \tag{5.44}$$

while the shift will be evolved using the hyperbolic "$\tilde{\Gamma}$-driver" equation [21],

$$\partial_t\ \beta^i = \frac{3}{4}B^i + \beta^j\partial_j\beta^i \tag{5.45}$$

$$\partial_t\ B^i = \partial_t\ \tilde{\Gamma}^i - \beta^j\partial_j\ \tilde{\Gamma}^i + \beta^j\partial_j B^i - \eta B^i \tag{5.46}$$

Where η is a parameter which acts as a damping coefficient, and it is typically set to values of the order of unity for the simulations that we will carry out.

Now we still need to find the conformal equivalents of (3.5) and (3.6). The first one can be easily transformed into:

$$\partial_t\ \theta = \frac{\alpha}{2}[R + 2\nabla_k Z^k + \tilde{A}_{ij}\tilde{A}^{ij} + \frac{2}{3}K^2 - 2K\theta] - Z^k\partial_k\alpha \tag{5.47}$$

And the conformal equivalent of (3.6) is

$$\partial_t\ Z_i = \alpha\nabla_j\ \tilde{A}^j_i - \frac{2}{3}\alpha\partial_i\ K + \alpha\partial_i\ \theta - 2\alpha Z_j\tilde{A}^j_i - \frac{2}{3}\alpha Z_i K - \theta\partial_i\ \alpha \tag{5.48}$$

Given our choice for the shift evolution equation, which involves the time derivative of $\tilde{\Gamma}^i$, we will also calculate the evolution equation for that quantity. If we use the definition of $\tilde{\Gamma}^i$, its evolution equation, and the evolution equation for Z_i in terms of the momentum constraint:

$$\tilde{\Gamma}^i = \tilde{\gamma}^{ir}\tilde{\gamma}^{jk}\tilde{\gamma}_{ij,k} \tag{5.49}$$

$$\partial_t\ \tilde{\Gamma}^i = -2(\alpha\partial_j\ \tilde{A}^{ij} + \tilde{A}^{ij}\partial_j\ \alpha) \tag{5.50}$$

$$\partial_t\ Z_i = \alpha M_i + \alpha\partial_i\ \theta - 2\alpha Z_j\tilde{A}^j_i - \frac{2}{3}\alpha K Z_i - \theta\partial_i\ \alpha \tag{5.51}$$

$$\tilde{M}^i = \tilde{\gamma}^{ij}M_j = \partial_j\ \tilde{A}^{ij} + \tilde{\Gamma}^i_{jk}\tilde{A}^{jk} - \frac{2}{3}\tilde{\gamma}^{ij}\partial_j\ K - \frac{\kappa}{2}\tilde{A}^{ij}\partial_j\ (log\chi) \tag{5.52}$$

Where M stands for the momentum constraint and $\tilde{\Gamma}^i_{jk}$ is the Christoffel symbol calculated with the conformal metric. We can substitute both (5.52 and 5.51) in (5.50) and we obtain:

$$
\begin{aligned}
\partial_t\ (\tilde{\Gamma}^i + 2\tilde{\gamma}^{ki}Z_k) &= 2Z_k\partial_t\ \tilde{\gamma}^{ki} + 2\alpha\tilde{\Gamma}^i_{jk}\tilde{A}^{jk} - \frac{4}{3}\alpha\tilde{\gamma}^{ij}\partial_j\ K - \kappa\alpha\tilde{A}^{ij}\partial_j\ (log\chi) \\
&\quad -2\tilde{A}^{ij}\partial_j\ \alpha + 2\tilde{\gamma}^{ki}(\alpha\partial_k\ \theta - 2\alpha Z_j A^j_k - \frac{2}{3}\alpha K Z_k - \theta\partial_k\ \alpha) \\
\partial_t\ \hat{\Gamma}^i &= 2\alpha\tilde{\Gamma}^i_{jk}\tilde{A}^{jk} - \frac{4}{3}\alpha\tilde{\gamma}^{ij}\partial_j\ K - \kappa\alpha\tilde{A}^{ij}\partial_j\ (log\chi) \\
&\quad -2\tilde{A}^{ij}\partial_j\ \alpha + 2\tilde{\gamma}^{ki}(\alpha\partial_k\ \theta - \frac{2}{3}\alpha K Z_k - \theta\partial_k\ \alpha)
\end{aligned}
\tag{5.53}
$$

Where we have defined $\hat{\Gamma}^i = \tilde{\Gamma}^i + 2\tilde{\gamma}^{ki}Z_k$. This is more or less (without some of the terms we used here) what BSSN is doing, substituting the momentum constraint in the evolution equation of $\tilde{\Gamma}^i$ and then setting the constraint to zero. BSSN is devised this way in order to ensure strong hyperbolicity. As we have just shown, this is equivalent (or partially equivalent if you drop terms) to evolve a different variable, $\hat{\Gamma}^i$. This is the reason why in most BSSN codes there is an explicit difference between what they call *local* $\tilde{\Gamma}^i$, which is calculated exclusively from the evolved conformal metric and the evolved variable, which is simply called $\tilde{\Gamma}^i$. We can then take advantage of the relationship between the two to calculate Z_i

$$
2Z_i = \tilde{\gamma}_{ij}\hat{\Gamma}^j - \tilde{\gamma}^{jk}\tilde{\gamma}_{ij,k} \tag{5.54}
$$

So, to sum up, the evolution equations for the conformal version of the Z4, including gauge choices, are:

$$\partial_t\ \alpha = -2\alpha(K - m\theta) + \beta^k \partial_k\ \alpha \tag{5.55}$$

$$\partial_t\ \beta^i = \frac{3}{4} B^i + \beta^k \partial_k \beta^i \tag{5.56}$$

$$\partial_t\ B^i = \partial_t\ \hat{\Gamma}^i - \beta^k \partial_k\ \hat{\Gamma}^i + \beta^k \partial_k B^i - \eta B^i \tag{5.57}$$

$$\partial_t\ \chi = \frac{2}{\kappa} \alpha \chi K + \beta^k \partial_k\ \chi - \frac{2}{\kappa} \chi \partial_k\ \beta^k \tag{5.58}$$

$$\partial_t\ \tilde{\gamma}_{ij} = -2\alpha \tilde{A}_{ij} + \beta^k \partial_k\ \tilde{\gamma}_{ij} + 2\tilde{\gamma}_{k(i} \partial_{j)}\ \beta^k - \frac{2}{3} \tilde{\gamma}_{ij} \partial_k\ \beta^k \tag{5.59}$$

$$\partial_t\ K = -\nabla^i \alpha_j + \alpha\ [R + 2\ \nabla^i Z_i + K^2 - 2\ \theta K] + \beta^k \partial_k\ K \tag{5.60}$$

$$\begin{aligned} \partial_t\ \tilde{A}_{ij} = {} & \chi^{\kappa/3} [-\nabla_i \alpha_j + \alpha(R_{ij} + \nabla_i Z_j + \nabla_j Z_i)]^{TF} + \alpha \tilde{A}_{ij}(K - 2\theta) \\ & - 2\alpha \tilde{A}_{il} \tilde{A}^l_j + \beta^k \partial_k\ \tilde{A}_{ij} + 2\tilde{A}_{k(i} \partial_{j)}\ \beta^k - \frac{2}{3} \tilde{A}_{ij} \partial_k\ \beta^k \qquad (5.61) \\ & + \partial_k\ Z_i \beta^k + Z_k \partial_i\ \beta^k \qquad (5.62) \end{aligned}$$

$$\partial_t\ \theta = \frac{\alpha}{2} [R + 2\nabla_k Z^k + \tilde{A}_{ij} \tilde{A}^{ij} + \frac{2}{3} K^2 - 2K\theta] - Z^k \partial_k\ \alpha + \beta^k \partial_k\ \theta \tag{5.63}$$

$$\begin{aligned} \partial_t\ \hat{\Gamma}^i = {} & 2\alpha \tilde{\Gamma}^i_{jk} \tilde{A}^{jk} - \frac{4}{3} \alpha \tilde{\gamma}^{ij} \partial_j\ K - \kappa \alpha \tilde{A}^{ij} \partial_j\ (log\chi) \\ & - 2\tilde{A}^{ij} \partial_j\ \alpha + 2\tilde{\gamma}^{ki} (\alpha \partial_k\ \theta - \frac{2}{3} \alpha K Z_k - \theta \partial_k\ \alpha) \\ & + \tilde{\gamma}^{kl} \partial_k\ \beta_l \beta^i + \frac{1}{3} \tilde{\gamma}^{ik} \partial_k \partial_l\ \beta^l - \hat{\Gamma}^k \partial_k\ \beta^i + \frac{2}{3} \hat{\Gamma}^i \partial_k\ \beta^k + \beta^k \partial_k\ \hat{\Gamma}^i \end{aligned} \tag{5.64}$$

Where we have included the shift terms that were omitted previously for the sake of clarity. For a justification of how we can add the shift terms after the calculation see for instance [8]. Differences with [27] are in the covariance of the Z-derivatives included in the evolution equations for the trace and trace-free parts of the extrinsic curvature, and also in the θ terms in those equations. The Z and $\theta\ \partial_k \alpha$ terms in the evolution equation for $\hat{\Gamma}$ are also missing in the reference [27]. Any arbitrariness regarding the use of $\hat{\Gamma}$ or $\tilde{\Gamma}$ in different parts of the code is also resolved here. During evolution there is also no need to enforce that $trA = 0$ whatsoever.

We can show here the results for a single black hole in 3D with puncture initial data. Evolution of the (precollapsed) lapse is shown and the freezing can be appreciated in Fig. 5.1.

5.6 Summary

We are proposing the action (5.1), which generalizes the Einstein-Hilbert one. Starting from this action one gets directly the Z4 field equations, plus the metric connection condition (which is to be enforced 'a posteriori' in our Palatini approach), plus the

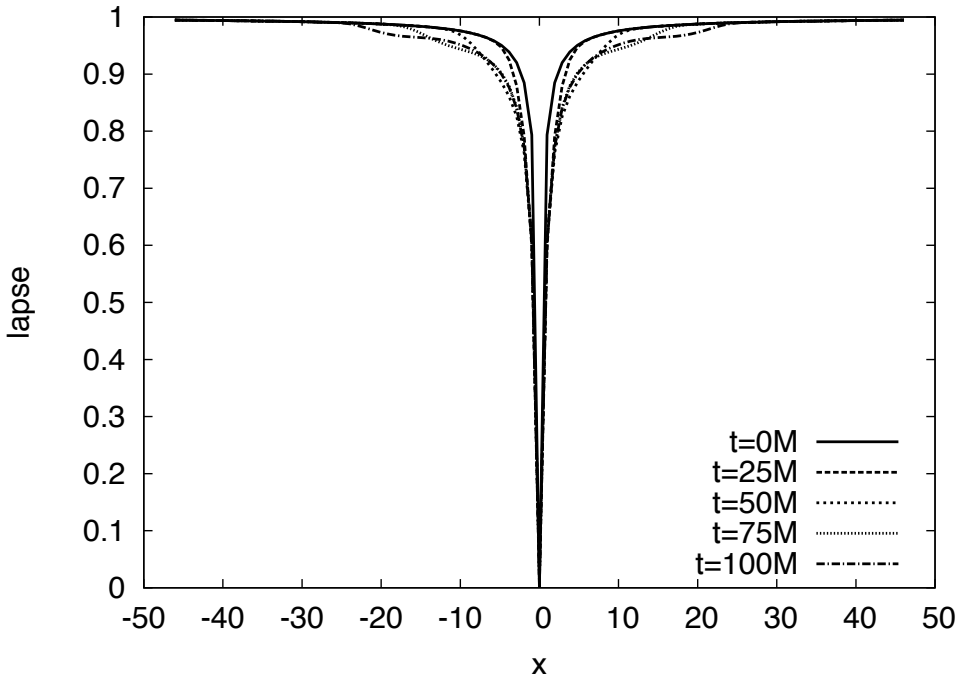


FIGURE 5.1: Plot of the lapse profile from $t = 0M$ to $t = 100M$ every $25M$. Simulation has been carried out with the Llama code. Freezing of the lapse profile can be clearly seen

constraints (5.12) stating the vanishing of Z_μ. We have shown how a suitable treatment of these constraints allows working with the Z4 covariant evolution in the way one usually does in numerical relativity. The price to pay for this general-covariant approach is that closing the evolution system requires a separate coordinate gauge specification. The challenge is then to incorporate the evolution equations for the gauge-related quantities (lapse and shift) into the canonical formalism, either via Lagrange multipliers [4, 5] or by any other means.

We have also presented an alternative strategy, based in the 'gauge sources' approach, which characterizes the generalized harmonic formalisms. This allows to dispose of the additional Z_μ vector field by enforcing at the same time the required coordinate conditions by means of some generalized gauge sources. The advantage of this second approach, at the price of getting a non-covariant evolution system, is that it can allow a direct use of symplectic integrators, devised to ensure constraint preservation during numerical evolution (see for instance ref. [6]). We have actually identified the gauge sources corresponding to some standard 3+1 gauge conditions, like the 'puncture gauge' consisting of the '1+log' lapse plus the gamma-driver shift prescriptions. The fact that these popular gauge conditions can fit into a Lagrangian/Hamiltonian approach, in the way we have shown, opens the door to new numerical relativity developments.

Finally we have devised a conformal decomposition for the Z4 system with all terms working for a puncture BH. There is now plenty of room to try other gauge choices (evolve Z and use gamma tilde, not gamma hat, in shift condition, etc.).

References

[1] Y. Choquet-Bruhat and T. Ruggeri, Comm. Math. Phys. **89**, 269 (1983).

[2] A. Anderson and J.W. York, Phys. Rev. Lett. **81**, 1154-1157 (1998).

[3] R. Arnowitt, S. Deser and C.W. Misner, in *Gravitation: an introduction to current research*, ed. L. Witten (Wiley: New York 1962). gr-qc/0405109.

[4] J. D. Brown in, *Quantum Mechanics of Fundamental Systems: The Quest for Beauty and Simplicity*, ed. by M. Henneaux and J. Zanelli (Springer, Berlin–Heidelberg–New York 2009). ArXiv:0803.0334 [gr-qc].

[5] D. Hilditch and R. Richter, arXiv:1002.4119v1 [gr-qc] (2010).

[6] C. Di Bartolo, R. Gambini and Jorge Pullin, J. Math. Phys. **46**, 032501 (2005).

[7] C. Bona, T. Ledvinka, C. Palenzuela and M. Žáček, Phys. Rev. D **69**, 064036 (2004).

[8] C. Bona, C. Palenzuela and C. Bona-Casas *Elements of Numerical Relativity and Relativistic Hydrodynamics*, Lect. Notes Phys. **783** (Springer, Berlin–Heidelberg–New York 2009).

[9] C. Bona, T. Ledvinka, C. Palenzuela and M. Žáček, Phys. Rev. D **67**, 104005 (2003).

[10] T. De Donder, *La Gravifique Einstenienne* Gauthier-Villars, Paris (1921).
The Mathematical Theory of Relativity, (Massachusetts Institute of Technology, 1927).

[11] K. Lanczos, Ann. Phys. **13**, 621 (1922).
Z. Phys. **23**, 537 (1923).

[12] Fock, V.A., *The Theory of Space, Time and Gravitation*, Pergamon, London (1959).

[13] Y. Choquet (Fourès)-Bruhat, Acta Matematica **88**, 141 (1952).
'Cauchy problem' in *Gravitation: An introduction to Current Research*, ed. by L. Witten, Wiley, (New York 1962).

[14] M. Shibata and T. Nakamura, Phys. Rev. D**52** 5428 (1995).

[15] T. W. Baumgarte and S. L. Shapiro, Phys. Rev. D**59** 024007 (1998).

[16] H. Friedrich, Comm. Math. Phys. **100**, 525(1985).

[17] F. Pretorius, Phys. Rev. Lett. **95** 121101 (2005).

[18] L. Lindblom et al, Class. Quantum Grav. **23**, S447 (2006).

[19] C. Bona, T. Ledvinka and C. Palenzuela, Phys. Rev. D **66**, 084013 (2002).

[20] F. Pretorius, Class. Quant. Grav. **23**, S529 (2006).

[21] M. Alcubierre, B. Brügmann, P. Diener, M. Koppitz, D. Pollney, E. Seidel and R. Takahashi Phys. Rev. D **67**, 084023 (2003)

[22] J. R. Van Meter, J. G. Baker, M. Koppitz and D-I. Choi, Phys. Rev. **D73**, 124001 (2006).

[23] C. Bona and J. Massó, Phys. Rev. Lett. **68** 1097 (1992)

[24] C. Bona, J. Massó, E. Seidel and J. Stela, Phys. Rev. Lett. **75** 600 (1995).

[25] A. Abrahams, A. Anderson, Y. Choquet-Bruhat and J. W. York, Phys. Rev. Lett. **75**, 3377 (1995).

[26] D. Alic, C. Bona and C. Bona-Casas, Phys. Rev. D **79**, 044026 (2009)

[27] S. Bernuzzi and D. Hilditch Phys. Rev. D **81**, 084003 (2010)

[28] M. Campanelli, C. O. Lousto, P. Marronetti and Y. Zlochower Phys. Rev. Lett. **96**, 111101 (2006)

[29] P. Marronetti, W. Tichy, B. Bruegmann, J. Gonzalez and U. Sperhake Phys. Rev. D **77**, 064010 (2008)

[30] C. Bona, J. Massó, E. Seidel and J. Stela, Phys. Rev. Lett. **75** 600 (1995).

Appendix A

Stability and Monotonicity

Let us assume that (the principal part of) the evolution system is strongly hyperbolic. This means that, for any chosen direction, we can express the system as a set of simple advection equations for the characteristic variables (eigenfields). In order to verify the stability properties of the proposed algorithms, it will be enough to consider a single advection equation with a generic speed v. The corresponding Flux will be given then by

$$F(u) = v\ u\ . \tag{A.1}$$

We will consider in the first place the first-order accurate approximation, obtained by a piecewise constant reconstruction (zero slope). The corresponding discretization can be obtained by replacing the prescription (1.26) into the general expression (1.10). The result is the linear three-point algorithm:

$$\begin{aligned} u_i^{n+1} = u_i^n \quad &+ \quad \frac{\Delta t}{\Delta x}\ [\ \frac{1}{2}\ (\lambda_{i+1} - v_{i+1})\ u_{i+1}^n \\ &+ \quad \frac{1}{2}\ (\lambda_{i-1} + v_{i-1})\ u_{i-1}^n - \lambda_i\ u_i^n\ ]\ . \end{aligned} \tag{A.2}$$

Allowing for the fact that λ is chosen at every point as the absolute value of the maximum speed, we can see that all the u^n coefficients are positive provided that the Courant stability condition

$$\lambda\ \frac{\Delta t}{\Delta x} \leq 1 \tag{A.3}$$

is satisfied. Note however that a more restrictive condition is obtained in the three-dimensional case, where we must add up in (A.2) the contributions from every space direction.

As it is well known, the positivity of all the coefficients ensures that the algorithm is monotonicity-preserving, so that spurious numerical oscillations can not appear. This

implies stability, but the converse is not true, as it is well known. Let us remember at this point that the centered FD discretization could be recovered from (A.2) simply by setting λ to zero, although we would lose the monotonicity property in this way.

The monotonicity properties of the piecewise constant reconstruction are not ensured in the piecewise linear case. We can clearly see in Fig. 1.1 that monotonicity problems can arise in steep gradient regions. The reason is that either the series of left $\{u^L\}$ or right $\{u^R\}$ interface predictions can show spurious peaks which where not present in the original function. In the case of the centered slope (1.12), a detailed analysis shows that this will happen at a given interface only if the left and right slopes differ by a factor of three or more. This gives a more precise sense to the 'steep gradient' notion in the centered slopes case.

The natural way to remedy this is to enforce that both (left and right) interface predictions are in the interval limited by the corresponding left and right point values (interwinding requirement). This amounts to using the 'limited' slopes

$$\sigma^{lim} = minmod(\ 2\sigma^L\ ,\ \sigma\ ,\ 2\sigma^R\ )\ , \tag{A.4}$$

where σ is the default slope at the given cell. This interwinding requirement is not enough, however, to ensure the positivity of all the coefficients in the resulting algorithm. A detailed analysis shows that an extra factor in the Courant condition would be required for monotonicity in this case:

$$\lambda\ \frac{\Delta t}{\Delta x} \leq 1/2\ . \tag{A.5}$$

Note however that we are analyzing here the elementary step (1.10). This is just the building block of the time evolution algorithm. The exact stability and monotonicity limits for the time step would depend on the specific choice of the full time evolution algorithm (see [6] from Chapter 1), which will be described in Appendix B.

A word of caution must be given at this point. It is well known that the monotonicity results hold only for strictly Flux-conservative algorithms. This is not our case: the Source terms play an important physical role. Of course, these terms do not belong to the principal part, so that positivity of the Flux terms ensures some strong form of stability. Nevertheless, one must be very careful with the physical interpretation, because the first-order constraints (1.5) preclude any clear-cut isolation of the Source terms. This makes the analogy with Fluid Dynamics only approximative and the use of the slope limiters a risky matter: we could be removing in the Flux part some features that are required to compensate something in the Source part. Our experience is that, at least for smooth profiles, more robust numerical simulations are obtained when the

slope limiters are switched off. The high frequency modes are kept under control by the numerical dissipation built in the proposed FV methods.

Appendix B

Time accuracy

The simple step (1.10) is only first-order accurate in time, and this fact is not changed by any of the space accuracy improvements we have considered up to now. The standard way of improving time accuracy is by the method of lines (MoL, see refs. [17] [6] in Chapter 1). The idea is to consider (1.10) as a basic evolution step

$$E(\ u^n\ ,\ \Delta t\ ) \tag{B.1}$$

in order to build higher order algorithms. A convenient choice for these time evolution algorithms is provided the standard Runge-Kutta methods [3] (see also [6]). For instance, second order accuracy can be obtained in two steps:

$$u^* = E(\ u^n,\ \Delta t\ ) \qquad u^{n+1} = \frac{1}{2}\ [\ u^n + E(\ u^*,\ \Delta t\ )\ ], \tag{B.2}$$

and third-order time accuracy with one more intermediate step:

$$\begin{aligned} u^{**} &= \frac{3}{4}\ u^n + \frac{1}{4}\ E(\ u^*\ ,\ \Delta t\ ) \\ u^{n+1} &= \frac{1}{3}\ u^n + \frac{2}{3}\ E(\ u^{**}\ ,\ \Delta t\ )\ . \end{aligned} \tag{B.3}$$

Note that the positivity of all the coefficients in (B.2, B.3) ensures that the monotonicity property of the basic step (B.1) will be preserved by the resulting strong-stability-preserving (SSP) algorithm. This interesting property comes at the price of keeping the upper limit on Δt that is required for the monotonicity of the basic step. This is a clear disadvantage with respect to the case in which the standard FD approach is being used for space discretization, in which one is only limited by plain stability, not monotonicity. Then, there are Runge-Kutta algorithms (with non-positive coefficients) that alow to

take Δt larger than the one required by the standard Courant condition (see [6] from Chapter 1).

Conversely, second order Runge-Kutta algorithms like (B.2) are unstable when used in combination with FD space discretization, unless artificial dissipation is added in order to recover stability, not just monotonicity (see [6] from Chapter 1). This is why FD simulations currently use at least a third-order time evolution algorithm.

Appendix C

Z3 evolution equations

The Z3 evolution system [11, 12] is given by:

$$(\partial_t \quad - \quad \mathcal{L}_\beta)\,\gamma_{ij} = -2\,\alpha\,K_{ij} \tag{C.1}$$

$$\begin{aligned}(\partial_t \quad - \quad \mathcal{L}_\beta)K_{ij} &= -\nabla_i\,\alpha_j + \alpha\,[\,R_{ij} + \nabla_i Z_j + \nabla_j Z_i \\ &- 2\,K_{ij}^2 + trK\,K_{ij} - S_{ij} + \frac{1}{2}\,(\,trS + (n-1)\,\tau\,)\,\gamma_{ij}\,] \\ &- \frac{n}{4}\,\alpha\,[\,tr\,R + 2\,\nabla_k Z^k \\ &\qquad +4\,tr^2K - tr(K^2) - 2\,Z^k\alpha_k/\alpha\,]\,\gamma_{ij}\end{aligned} \tag{C.2}$$

$$(\partial_t \quad - \quad \mathcal{L}_\beta)Z_i = \alpha\,[\nabla_j\,(K_i{}^j - \delta_i{}^j\;trK) - 2K_i{}^j Z_j - S_i]\;, \tag{C.3}$$

where n is an arbitrary parameter governing the coupling of the energy constraint.

The fully first-order version can be obtained in the standard way, by introducing the additional fields

$$D_{kij} \equiv \frac{1}{2}\;\partial_k\;\gamma_{ij}\;. \tag{C.4}$$

Note that the ordering constraint (1.5) reads

$$\partial_r\;D_{kij} = \partial_k\;D_{rij}\;, \tag{C.5}$$

which is no longer an identity for the first order system. As a consequence of this ordering ambiguity of second derivatives, the Ricci tensor term in (the first order version of) the evolution equation (C.2) can be written in many different ways. Then, an ordering parameter ζ can be introduced [12], so that the parameter choice $\zeta = +1$ corresponds to the standard Ricci decomposition

$${}^{(3)}R_{ij} \;=\; \partial_k\,\Gamma^k{}_{ij} - \partial_i\,\Gamma^k{}_{kj} + \Gamma^r{}_{rk}\Gamma^k{}_{ij} - \Gamma^k{}_{ri}\Gamma^r{}_{kj} \tag{C.6}$$

whereas the opposite choice $\zeta = -1$ corresponds instead to the decomposition

$$\begin{aligned} {}^{(3)}R_{ij} &= -\partial_k\, D^k{}_{ij} + \partial_{(i}\,\Gamma_{j)k}{}^{k} - 2D_r{}^{rk} D_{kij} \\ &+ 4D^{rs}{}_i D_{rsj} - \Gamma_{irs}\Gamma_j{}^{rs} - \Gamma_{rij}\Gamma^{rk}{}_k \ , \end{aligned} \tag{C.7}$$

which is most commonly used in Numerical Relativity codes. We can then consider the generic case as a linear combination of (C.6) and (C.7).

In the spherically symmetric vacuum case, the first order version of the system (C.1-C.2) is free of any ordering ambiguity. It can be written as

$$\partial_t\,\gamma_{rr} = -2\,\alpha\,\gamma_{rr}\,K_r{}^r, \qquad \partial_t\,\gamma_{\theta\theta} = -2\,\alpha\,\gamma_{\theta\theta}\,K_\theta{}^\theta \tag{C.8}$$

$$\begin{aligned} \partial_t\,K_r{}^r &+ \partial_r[\alpha\,\gamma^{rr}\,(A_r + (2-n)\,D_\theta{}^\theta - (2-n/2)\,Z_r)] = \\ &\alpha\,[(K_r{}^r)^2 + (2-n)\,K_r{}^r\,K_\theta{}^\theta - (n/2)\,(K_\theta{}^\theta)^2 \\ &-\gamma^{rr}\,D_r{}^r\,(A_r + (2-n)\,D_\theta{}^\theta + (n/2-2)\,Z_r) \\ &+\gamma^{rr}\,D_\theta{}^\theta\,((2-n)\,A_r - (2-3\,n/2)\,D_\theta{}^\theta - n\,Z_r) \\ &-\gamma^{rr}\,(2-n)\,A_r\,Z_r - (n/2)\,\gamma^{\theta\theta}] \end{aligned} \tag{C.9}$$

$$\begin{aligned} \partial_t\,K_\theta{}^\theta &+ \partial_r[\alpha\,\gamma^{rr}\,((1-n)\,D_\theta{}^\theta + (n/2)\,Z_r)] = \\ &\alpha\,[(1-n)\,K_r{}^r K_\theta{}^\theta + (2-n/2)\,(K_\theta{}^\theta)^2 \\ &-\gamma^{rr}\,D_r{}^r\,((1-n)\,D_\theta{}^\theta + (n/2)\,Z_r) \\ &+\gamma^{rr}\,D_\theta{}^\theta\,((2-n)\,Z_r - (2-3n/2)\,D_\theta{}^\theta) \\ &-n\,\gamma^{rr}\,A_r\,(D_\theta{}^\theta - Z_r) + (1-n/2)\,\gamma^{\theta\theta}] \end{aligned} \tag{C.10}$$

$$\begin{aligned} \partial_t\,Z_r &+ \partial_r[2\,\alpha\,K_\theta{}^\theta] = \\ &2\,\alpha\,[D_\theta{}^\theta\,(K_r{}^r - K_\theta{}^\theta) + A_r\,K_\theta{}^\theta - K_r{}^r\,Z_r] \end{aligned} \tag{C.11}$$

$$\partial_t\,D_r{}^r + \partial_r[\alpha\,K_r{}^r] = 0, \qquad \partial_t\,D_\theta{}^\theta + \partial_r[\alpha\,K_\theta{}^\theta] = 0, \tag{C.12}$$

where we are using normal coordinates (zero shift). The slicing condition (1.44) can be written as

$$\partial_t\,\alpha = -\alpha^2\,f\,trK\ , \qquad \partial_t\,A_r + \partial_r[\alpha\,f\,trK] = 0\ . \tag{C.13}$$

The mass function can be defined for spherically symmetric spacetimes as [14]

$$2M = Y\ [\ 1 - g^{ab}\partial_a\,Y\,\partial_b\,Y\ ]\ , \tag{C.14}$$

where Y stands for the area radius. In spherical coordinates we get

$$2M(t,r) = \sqrt{\gamma_{\theta\theta}}\ \{\ 1 + \gamma_{\theta\theta}\,[(K_\theta{}^\theta)^2 - \gamma^{rr}(D_\theta{}^\theta)^2]\ \}\ . \tag{C.15}$$

The mass function has a clear physical interpretation: it provides the mass inside a sphere of radius r at the given time t. It follows that $M(t, r)$ must be constant for the Schwarzschild spacetime, no matter which coordinates are being used. This provides a convenient accuracy check for numerical simulations.

Appendix D

Hyperbolicity of the adjusted first-order Z4 system

We will write the first-order evolution system in a balance-law form. For a generic quantity u, this leads to

$$\partial_t\ u + \partial_k\ F^k(u) = S(u)\ , \tag{D.1}$$

where the Flux $F^k(u)$ and Source terms $S(u)$ can depend on the full set of dynamical fields in an algebraic way. In the case of the space-derivatives fields, their evolution equations (3.15-3.17) are yet in the balance-law form (D.1). Note however that any damping terms of the form described in (3.18) will contribute both to the Flux and the Source terms in a simple way.

The metric evolution equation (3.3) will be written in the form

$$\partial_t\ \gamma_{ij} = 2\,\beta^k D_{kij} + B_{ij} + B_{j\,i} - 2\ \alpha\ K_{ij}\ , \tag{D.2}$$

so that it is free of any Flux terms. The remaining (non-trivial) evolution equations (3.4- 3.6) require a more detailed development. We will expand first the Flux terms in the following way:

$$\partial_t K_{ij} \quad + \quad \partial_k[-\beta^k\ K_{ij} + \alpha\ \lambda^k{}_{ij}\ ] = S(K_{ij}) \tag{D.3}$$

$$\partial_t Z_i \quad + \quad \partial_k[-\beta^k\ Z_i + \alpha\ \{-K^k{}_i + \delta^k{}_i(trK - \Theta)\} \tag{D.4}$$

$$+\mu\ (B_i{}^k - \delta_i{}^k trB)\ ] = S(Z_i)$$

$$\partial_t\Theta \quad + \quad \partial_k[-\beta^k\ \Theta + \alpha\ (D^k - E^k - Z^k)\ ] = S(\Theta) \tag{D.5}$$

where we have used the shortcuts $D_i \equiv D_{ik}{}^k$ and $E_i \equiv D^k{}_{ki}$, and

$$\begin{aligned}\lambda^k{}_{ij} = D^k{}_{ij} \quad &- \quad \frac{1}{2}\,(1+\xi)\,(D_{ij}{}^k + D_{j\,i}{}^k) \\ &+ \quad \frac{1}{2}\,\delta^k{}_i\,[A_j + D_j - (1-\xi)\,E_j - 2\,Z_j\,] \\ &+ \quad \frac{1}{2}\,\delta^k{}_j\,[A_i + D_i - (1-\xi)\,E_i - 2\,Z_i\,]\,. \end{aligned} \tag{D.6}$$

The Source terms $S(u)$ do not belong to the principal part and will be displayed later. Let us focus for the moment in the hyperbolicity analysis, by selecting a specific space direction $\vec{n}$, so that the corresponding characteristic matrix is

$$A^n = \frac{\partial\, F^n}{\partial\, u}\,, \tag{D.7}$$

where the symbol n replacing an index stands for the projection along the selected direction $\vec{n}$. We can get by inspection the following (partial) set of eigenfields, independently of the gauge choice:

- **Transverse derivatives**:

$$A_\perp\,, \qquad B_\perp{}^i\,, \qquad D_{\perp ij}\,, \tag{D.8}$$

  propagating along the normal lines (characteristic speed $-\beta^n$). The symbol $\perp$ replacing an index means the projection orthogonal to $\vec{n}$.

- **Light-cone eigenfields**, given by the pairs

$$F^n[\,D_{n\perp\perp}\,] \quad \pm \quad F^n[\,K_{\perp\perp}\,] \tag{D.9}$$

$$-F^n[\,Z_\perp\,] \quad \pm \quad F^n[\,K_{n\perp}\,] \tag{D.10}$$

$$F^n[\,D_n - E_n - Z_n\,] \quad \pm \quad F^n[\,\Theta\,] \tag{D.11}$$

  with characteristic speed $-\beta^n \pm \alpha$, respectively.

Note that the eigenvector expressions given above, in terms of the Fluxes, are valid for any choice of the ordering parameters μ and ξ. Only the detailed expression of the eigenvectors, obtained from the Flux definitions, is affected by these parameter choices. For instance

$$F^n[\,D_n - E_n - Z_n\,] = -\beta^n\,[\,D_n - E_n - Z_n\,] + \alpha\,\theta + (\mu - 1)\,tr(B_{\perp\perp})\,. \tag{D.12}$$

Any value $\mu \neq 1$ implies that the characteristic matrix of the Z3 system (see Appendix C and Chapter 1), obtained by removing the variable θ from our Z4 evolution system

(see [19] from Chapter 3), can not be fully diagonalized in the dynamical shift case. Of course, the hyperbolicity analysis can not be completed until we get suitable coordinate conditions, amounting to some prescription for the lapse and shift sources Q and Q_i, respectively. But the subset of eigenvectors given here is gauge independent: non-diagonal blocs can not be fixed *a posteriori* by the coordinates choice.

The detailed expressions for the eigenvectors can be relevant when trying to compare with related formulations. For instance, a straightforward calculation shows that the eigenvectors (D.9-D.11) can be matched to the corresponding ones in the harmonic formalism if and only if

$$\xi = -1\,, \qquad \mu = 1/2\,. \tag{D.13}$$

This shows that different requirements can point to different choices of these ordering parameters. We prefer then to leave this choice open for future applications. Concerning the simulations in this paper, we have taken $\xi = -1\,,\ \mu = 1\,.$

Finally, we give for completeness the Source terms, namely:

$$\begin{aligned}
S(K_{ij}) &= -K_{ij}\ trB + K_{ik}\ B_j{}^k + K_{jk}\ B_i{}^k + \alpha\ \{\frac{1}{2}\ (1+\xi)\ [-A_k\ \Gamma^k{}_{ij} + \frac{1}{2}(A_i\ D_j + A_j\ D_i)] \\
&+ \frac{1}{2}\ (1-\xi)\ [A_k\ D^k{}_{ij} - \frac{1}{2}\{A_j\ (2\ E_i - D_i) + A_i\ (2\ E_j - D_j)\} \\
&+ 2\ (D_{ir}{}^m\ D^r{}_{mj} + D_{jr}{}^m\ D^r{}_{mi}) - 2\ E_k\ (D_{ij}{}^k + D_{ji}{}^k)] \\
&+ (D_k + A_k - 2\ Z_k)\ \Gamma^k{}_{ij} - \Gamma^k{}_{mj}\ \Gamma^m{}_{ki} - (A_i\ Z_j + A_j\ Z_i) - 2\ K^k{}_i\ K_{kj} \\
&+ (trK - 2\ \Theta)\ K_{ij}\} - 8\ \pi\ \alpha\ [S_{ij} - \frac{1}{2}\ (trS - \tau)\ \gamma_{ij}] \qquad \text{(D.14)} \\
S(Z_i) &= -Z_i\ trB + Z_k\ B_i{}^k + \alpha\ [A_i\ (trK - 2\ \Theta) - A_k\ K^k{}_i - K^k{}_r\ \Gamma^r{}_{ki} + K^k{}_i\ (D_k - 2\ Z_k)] \\
&\qquad -8\ \pi\ \alpha\ S_i \qquad \text{(D.15)} \\
S(\Theta) &= -\Theta\ trB + \frac{\alpha}{2}[2\ A_k\ (D^k - E^k - 2\ Z^k) + D_k{}^{rs}\ \Gamma^k{}_{rs} - D^k(D_k - 2\ Z_k) - K^k{}_r\ K^r{}_k \\
&+ trK\ (trK - 2\ \Theta)] - 8\ \pi\ \alpha\ \tau\ . \qquad \text{(D.16)}
\end{aligned}$$

Appendix E

Scalar field stuffing

Let us consider the stress-energy tensor

$$T_{ab} = \Phi_a \, \Phi_b - 1/2 \, (g^{cd} \Phi_c \, \Phi_d) \, g_{ab} \, , \tag{E.1}$$

where we have noted $\Phi_a = \partial_a \, \Phi$, corresponding to a scalar field matter content. The 3+1 decomposition of (E.1) is given by

$$\tau = 1/2 \, ({\Phi_n}^2 + \gamma^{kl} \Phi_k \, \Phi_l) \, , \qquad S_i = \Phi_n \, \Phi_i \, , \qquad S_{ij} = \Phi_i \, \Phi_j + 1/2 \, ({\Phi_n}^2 - \gamma^{kl} \Phi_k \, \Phi_l) \, \gamma_{ij} \, , \tag{E.2}$$

where Φ_n stands for the normal time derivative:

$$(\partial_t - \beta^k \, \partial_k) \, \Phi = -\alpha \, \Phi_n \, . \tag{E.3}$$

The quantities (E.2) appear as source terms in the field equations (C.2-C.3 in the Z3 case,3.4-3.6 in the Z4 case).

The stress-energy conservation amounts to the evolution equation for the scalar field, which is just the scalar wave equation. In the 3+1 language, it translates into the Flux-conservative form:

$$\partial_t \, [\, \sqrt{\gamma} \, \Phi_n \,] + \partial_k \, [\, \sqrt{\gamma} \, (-\beta^k \Phi_n + \alpha \, \gamma^{kj} \Phi_j) \,] = 0 \, . \tag{E.4}$$

A fully first-order system may be obtained by considering the space derivatives Φ_i as independent dynamical fields, as we did for the metric space derivatives.

Concerning the initial data, we must solve the energy-momentum constraints. They can be obtained by setting both Θ and Z_i to zero in (3.5, 3.6 in the Z4 case, C.3 in the Z3

case). In the time-symmetric case ($K_{ij} = 0$), this amounts to

$$R = 16\pi\,\tau\,, \qquad S_i = \Phi_n\,\Phi_i = 0\,. \tag{E.5}$$

The momentum constraint will be satisfied by taking Φ (and then Φ_i) to be zero everywhere on the initial time slice. Concerning the energy constraint, we will consider the line element (3.31) with $m = m(r)$. We assume a constant mass value $m = M$ for the black-hole exterior, so that the energy constraint in (E.5) will be satisfied with $\tau = 0$ there.

In the interior region, the energy constraint will translate instead into the equation

$$m'' = -2\pi r\ (\Phi_n)^2\ (1 + \frac{m}{2r})^5\ , \tag{E.6}$$

which can be interpreted as providing the initial Φ_n value for any convex ($m'' \leq 0$) mass profile. Of course, some regularity conditions both at the center and at the matching point r_0 must be assumed. Allowing for (E.6), we have taken

$$\begin{aligned} & m = m'' &=\ 0 \qquad & (r = 0) \\ m = M, \qquad & m' = m'' &=\ 0 \qquad & (r = r_0)\ . \end{aligned}$$

Note that, allowing for (E.6), these matching conditions ensure just the continuity of Φ_n, not its smoothness. This can cause some numerical error, as we are currently evolving Φ_n through the differential equation (E.4). If this is a problem, we can demand the vanishing of additional derivatives of the mass function $m(r)$, both at the origin and at the matching point (this is actually the case in our shift simulations). This is not required in the standard case ($f = 2/\alpha$, normal coordinates), where we have used a simple profile, with the matching point at the apparent horizon ($r_0 = M/2$), given by

$$m(r) = 4r - 4/M[\,r^2 + (M/2\pi)^2\ sin^2(2\pi r/M)\,]\ . \tag{E.7}$$

Appendix F

Symmetric hyperbolicity of the Z4 system

We have derived in section II a generalized 'energy estimate' for the Z4 system, namely:

$$S = \Theta^2 + V_k V^k + \Pi^{ij}\Pi_{ij} + \tilde{\mu}^{kij}\tilde{\mu}_{kij} + (1+\zeta)(Z^k Z_k - \tilde{\mu}^{kij}\tilde{\mu}_{ijk}) + 2\,\zeta\, Z_k W^k\,, \qquad \text{(F.1)}$$

where we noted

$$\tilde{\mu}_{kij} = \mu_{kij} - W_k\,\gamma_{ij}\,. \qquad \text{(F.2)}$$

In order to check the positivity of (F.1), let us consider the decomposition of the three-index tensor $\tilde{\mu}_{kij}$ into its symmetric and antisymmetric parts, that is

$$\tilde{\mu}_{kij} = \tilde{\mu}_{(kij)} + \tilde{\mu}^{(a)}_{kij}\,. \qquad \text{(F.3)}$$

Allowing for the identities,

$$\tilde{\mu}^{(a)}_{(kij)} = 0\,, \qquad \tilde{\mu}^{(a)}_{(ij)k} = -\frac{1}{2}\,\tilde{\mu}^{(a)}_{kij}\,, \qquad \text{(F.4)}$$

the rank-three terms contribution to S can the be written as

$$S = -\,\zeta\;\tilde{\mu}_{(kij)}\;\tilde{\mu}^{(kij)} + \frac{3+\zeta}{2}\;\tilde{\mu}^{(a)}_{kij}\;\tilde{\mu}^{(a)\,kij} + \cdots \qquad \text{(F.5)}$$

(the dots stand for lower-rank components). It follows that a necessary condition for positivity is $0 \geq \zeta \geq -3$.

We can now rewrite (F.1) as

$$S = \Theta^2 + V_k V^k + \Pi^{ij}\,\Pi_{ij} - \zeta\;\tilde{\mu}_{kij}\;\tilde{\mu}^{kij} + \frac{3}{2}\;(1+\zeta)\;\tilde{\mu}^{(a)}_{kij}\;\tilde{\mu}^{(a)\;kij} + (1+\zeta)\;Z^k Z_k + 2\,\zeta\,Z_k W^k. \tag{F.6}$$

Allowing for (F.2), which implies in turn

$$\tilde{\mu}^{k}{}_{ki} = -Z_i - W_i\,, \tag{F.7}$$

we see that we can rewrite again (F.6) as

$$S = \Theta^2 + V_k V^k + \Pi^{ij}\Pi_{ij} - \zeta\;\tilde{\lambda}_{kij}\;\tilde{\lambda}^{kij} + \frac{3}{2}\;(1+\zeta)\;\tilde{\mu}^{(a)}_{kij}\;\tilde{\mu}^{(a)\;kij} + (1+\zeta)\;Z^k Z_k\,, \tag{F.8}$$

where

$$\tilde{\lambda}_{kij} = \lambda_{kij}\,|_{\zeta=-1} = \tilde{\mu}_{kij} + \gamma_{k(i}W_{j)}\,. \tag{F.9}$$

It follows from the final expression (F.8) that the energy estimate is positive definite in the whole interval

$$0 \;\geq\; \zeta \;\geq\; -1\,. \tag{F.10}$$

Note that for $\zeta = -1$, that is $\lambda = \tilde{\lambda}$, we recover the estimate given in ref. [9]. This confirms that Z4 in normal coordinates with harmonic slicing is symmetric hyperbolic for the range (F.10) of the ordering parameter.

Appendix G

Hyperbolicity of the energy modes

We can analyze the hyperbolicity of the boundary evolution system, by considering the characteristic matrix along a generic oblique direction $\mathbf{r}$, which is related to the normal direction $\mathbf{n}$ by

$$\mathbf{r} = \mathbf{n}\ cos\varphi + \mathbf{s}\ sin\varphi\ , \tag{G.1}$$

where we have taken

$$\mathbf{n}^2 = \mathbf{s}^2 = 1 \qquad \mathbf{n}\cdot\mathbf{s} = 0\ . \tag{G.2}$$

The strong hyperbolicity requirement amounts to demand that the characteristic matrix is fully diagonalizable and has real eigenvalues (propagation speeds) for any value of the angle φ.

In order to compute the characteristic matrix, we will consider the standard form of (the principal part of) the evolution system as follows

$$\partial_t\ \mathbf{u} + \alpha\,\partial_r\,\mathbf{F}^r(\mathbf{u}) = \cdots\ , \tag{G.3}$$

where $\mathbf{u}$ stands for the array of dynamical fields and $\mathbf{F}^r$ is the array of fluxes along the direction $\mathbf{r}$. We will restrict ourselves here to the Energy-modes subsystem, which consists in the fields

$$\mathbf{u} = (E^+\ ,\ E^-\ ,\ V_s\ ,\ V_p\ ) \tag{G.4}$$

the index p meaning here a projection along the direction orthogonal both to $\mathbf{n}$ and $\mathbf{s}$.

It is clear that the two components V_p are eigenvectors of the characteristic matrix with zero propagation speed. The non-trivial fluxes are then:

$$F^r(E^+) = cos\varphi\ E^+ + sin\varphi\ V_s \tag{G.5}$$

$$F^r(E^-) = (a-1)\,cos\varphi\ E^- + (1-a)\,sin\varphi\ V_s \tag{G.6}$$

$$F^r(V_s) = \frac{1}{2}\ sin\varphi\ (E^+ + E^-)\,, \tag{G.7}$$

where we have allowed for the modified evolution equation (4.38). We can see that the one of the characteristic speeds is zero and the other two are be given by the solutions of

$$(v - \alpha\, cos\varphi)\ (v - (a-1)\,\alpha\, cos\varphi) = (1 - \frac{a}{2})\ \alpha^2\ sin^2\varphi\ , \tag{G.8}$$

which has real distinct solutions for $a < 2$. The degenerate case $a = 2$ is not diagonalizable. It follows that the boundary evolution subsystem given by the above fluxes is strongly hyperbolic for $a < 2$ and weakly hyperbolic for $a = 2$.